TP 13437E

# ICE NAVIGATION SIMULATION – SHIPBOARD RADAR SIMULATION MODULE (SRSM)

**Prepared for**
Transportation Development Centre
Transport Canada

**by**
PhiloSoft Inc.
3608 Boulevard St. Charles, Suite #27c
Kirkland, Quebec
H9H 3C3

May 1999

TP 13437E

# ICE NAVIGATION SIMULATION – SHIPBOARD RADAR SIMULATION MODULE (SRSM)

by
Charles G. Marton
PhiloSoft Inc.

May 1999

This report reflects the views of PhiloSoft Inc. and not necessarily those of the Transportation Development Centre of Transport Canada or the sponsoring organizations.

The Transportation Development Centre does not endorse products or manufacturers. Trade or manufacturers' names appear in this report only because they are essential to its objectives.

Un sommaire français se trouve avant la table des matières.

Transport Canada / Transports Canada

# PUBLICATION DATA FORM

| 1. Transport Canada Publication No. | 2. Project No. | 3. Recipient's Catalogue No. |
|---|---|---|
| TP 13437E | 9127 | |

| 4. Title and Subtitle | 5. Publication Date |
|---|---|
| Ice Navigation Simulation – Shipboard Radar Simulation Module (SRSM) | May 1999 |
| | 6. Performing Organization Document No. |

| 7. Author(s) | 8. Transport Canada File No. |
|---|---|
| Charles G. Marton | ZCD1460-368 |

| 9. Performing Organization Name and Address | 10. PWGSC File No. |
|---|---|
| PhiloSoft Inc.<br>3608 St. Charles Blvd., #27c<br>Kirkland, Quebec<br>H9H 3C3 | XSD-6-01870 |
| | 11. PWGSC or Transport Canada Contract No. |
| | T8200-6-6550/A |

| 12. Sponsoring Agency Name and Address | 13. Type of Publication and Period Covered |
|---|---|
| Transportation Development Centre (TDC)<br>800 René Lévesque Blvd. West<br>Suite 600<br>Montreal, Quebec<br>H3B 1X9 | Final |
| | 14. Project Officer |
| | Charles Gautier |

15. Supplementary Notes (Funding programs, titles of related publications, etc.)

Co-sponsored by Prairie and Northern Region and the Program of Energy Research and Development (PERD)

16. Abstract


Marine radar is an integral part of the solution to the problems arising from ships navigating in ice-infested waters. This has been proven in various field trials sponsored by the Transportation Development Centre (TDC).

Phase 1 work demonstrated the feasibility of using synthetic aperture radar as a basis for simulating shipboard marine radar. The simulated radar was developed in conjunction with the larger ice navigation simulator platform.

The radar simulation software provides several key elements:

- range and azimuth blurring
- range attenuation
- shadow generation
- variable range setting
- network operation
- compatibility with existing commercial radar software interfaces

The radar simulator was successfully integrated with an ice navigation client application.


| 17. Key Words | 18. Distribution Statement |
|---|---|
| Marine, radar, simulation, ice navigation | Limited number of copies available from the Transportation Development Centre |

| 19. Security Classification (of this publication) | 20. Security Classification (of this page) | 21. Declassification (date) | 22. No. of Pages | 23. Price |
|---|---|---|---|---|
| Unclassified | Unclassified | — | xii, 32, apps | Shipping/ Handling |

Transports Canada | Transport Canada

# FORMULE DE DONNÉES POUR PUBLICATION

| 1. N° de la publication de Transports Canada | 2. N° de l'étude | 3. N° de catalogue du destinataire |
|---|---|---|
| TP 13437E | 9127 | |

| 4. Titre et sous-titre | 5. Date de la publication |
|---|---|
| Ice Navigation Simulation – Shipboard Radar Simulation Module (SRSM) | Mai 1999 |
| | 6. N° de document de l'organisme exécutant |

| 7. Auteur(s) | 8. N° de dossier - Transports Canada |
|---|---|
| Charles G. Marton | ZCD1460-368 |

| 9. Nom et adresse de l'organisme exécutant | 10. N° de dossier - TPSGC |
|---|---|
| PhiloSoft Inc.<br>3608, boul. Saint-Charles, #27c<br>Kirkland, Quebec<br>H9H 3C3 | XSD-6-01870 |
| | 11. N° de contrat - TPSGC ou Transports Canada |
| | T8200-6-6550/A |

| 12. Nom et adresse de l'organisme parrain | 13. Genre de publication et période visée |
|---|---|
| Centre de développement des transports (CDT)<br>800, boul. René-Lévesque Ouest<br>Bureau 600<br>Montréal (Québec)<br>H3B 1X9 | Final |
| | 14. Agent de projet |
| | Charles Gautier |

15. Remarques additionnelles (programmes de financement, titres de publications connexes, etc.)

Projet coparrainé par la région des Prairies et du Nord ainsi que le Programme de recherche et développement énergétiques (PRDE)

16. Résumé


Les radars de marine sont partie intégrante de la solution des problèmes de navigation maritime dans les eaux envahies par les glaces. Leur efficacité a été vérifiée dans divers essais sur le terrain organisés par le Centre de développement des transports (CDT).

Les travaux de la phase 1 ont démontré la faisabilité d'utiliser un radar à ouverture synthétique (ROS) pour simuler un radar de marine de bord. Le radar simulé a été élaboré conjointement avec la plate-forme de simulation de navigation dans les glaces.

Le logiciel de simulation du radar comporte les fonctions clés suivantes :

- brouillage de la distance et de l'azimut
- réduction de la distance
- production d'ombres
- réglage de la portée
- fonctionnement en réseau
- comptabilité avec les logiciels d'interface radar commerciaux actuels

Le simulateur radar a été intégré avec succès à une application client de navigation dans les glaces.


| 17. Mots clés | 18. Diffusion |
|---|---|
| Marine, radar, simulation, navigation dans les glaces | Le Centre de développement des transports dispose d'un nombre limité d'exemplaires. |

| 19. Classification de sécurité (de cette publication) | 20. Classification de sécurité (de cette page) | 21. Déclassification (date) | 22. Nombre de pages | 23. Prix |
|---|---|---|---|---|
| Non classifiée | Non classifiée | — | xii, 32, ann. | Port et manutention |

## ACKNOWLEDGEMENTS

A number of individuals contributed to the completion of this phase of the project. I would like to extend *extra special* thanks to Charles Gautier of the Transportation Development Centre for his many valuable suggestions and thorough review of this and related documentation.

Blank page

## EXECUTIVE SUMMARY

This project comes as a result of a study conducted on behalf of the Canadian Coast Guard (TP 12496) that outlined the essential elements required to make up an International Ice Navigator Course. A significant component of this course involved training ice navigators with the aid of a simulator. The need for a simulator has been recognized by such countries as Finland, Russia, Sweden, Germany and Norway.

Different approaches have been considered for the implementation of this capability, hereafter called the Ice Navigation Simulator, and a number of Canadian facilities exist where large modifiable simulation engines could be used to perform this task. However, these systems use proprietary technologies and are not considered "open" systems. Besides being very expensive, they do not allow for widespread distribution and availability. Given state-of-the-art PC technology, complete with near workstation performance at a fraction of the cost, together with the explosion of multimedia and available virtual reality equipment, it is now possible to consider alternatives to the fixed, high-cost and proprietary systems currently in use for shipboard simulation. Therefore, a new approach to a realistic, low-cost implementation of the simulator was adopted.

The simulator encompasses the following elements:

- Simulation of ice on shipboard radar
- Simulation and management of remotely sensed data
- Simulation of visual aspect of ice, in daylight and night transit conditions
- Ship transit simulation (basic at this stage)
- Ice recognition and ice climatology training aids
- Ice regime entry rules training aid
- Ice Navigation systems and Electronic Charting and Display Information System (ECDIS) support

Part of this simulation facility would require a Shipboard Radar Simulation Module (SRSM) to provide the data for training an operator for ice navigation. One of the critical elements in ice navigation is the use of marine radar. Therefore, a simulation of marine radar would be important and necessary in order to derive maximum benefit from the simulator.

Synthetic Aperture Radar (SAR) data, such as that taken by airborne SAR or even from a satellite such as RADARSAT, represents a very accurate picture of a ship's ice environment. When SAR imagery is collected, the spatial geometry and features in the imagery are accurate and the features readily identifiable. In most cases, features visible on the shipboard radar will correspond to equivalent features in the SAR imagery. In fact, the correlation is so strong that it forms the basis for locating a vessel relative to equivalent features in the SAR imagery. SAR data was used as the scenario base in the simulator. This project therefore dealt with using the SAR data together with radar simulation modelling to develop the radar simulation component of the Ice Navigation Simulator.

The overall program objective was to develop a low-cost, PC-based Ice Navigation Simulation platform to train entry-level ice navigators. The specific objective was to develop a simulation module of generic shipboard radar for use with the main simulation platform.

The SRSM gets its SAR data from the Main Simulator Platform (MSP) and provides simulated radar images to its clients. The SRSM uses image convolutions of SAR data, together with some off-line data interpretation, as the basis for simulation.

The radar simulation software provides several key elements:

- range and azimuth blurring
- range attenuation
- shadow generation
- variable range setting
- network operation
- compatibility with existing commercial radar software interfaces

The MSP provides the area-of-interest data (both SAR raster and height bitmap data) to the SRSM when requested. The SRSM then takes this data and converts it to the polar coordinate domain, where it performs range and azimuth blurring based on beam angle and pulse length. Range attenuation is applied, as required, by the radar settings. Finally, the radar shadows are generated in real time to create a shadow mask that is integrated with the radar image. At this point, an SRSM client can request data.

This phase of the project was a success, despite some initial technical difficulties with system performance and speed. The SRSM is operational, but much work remains to be done. This work includes, but is not limited to, the following:

1. Adding the ship's track in the simulated radar image
2. Performing pixel interpolation at lower range settings
3. Adding X-polarized image synthesis
4. Improving sea clutter simulation for open water
5. Adding a stand-alone image display
6. Adding on-line tuning of simulation
7. Improving radar shadow height map generation

## SOMMAIRE

Le projet est le résultat d'une étude menée au nom de la Garde côtière canadienne (TP 12496) qui a exposé les principaux éléments requis pour établir un cours international de navigation dans les glaces. La formation des navigateurs à l'aide d'un simulateur a été l'une des principales composantes de ce cours. Le besoin d'un simulateur a été reconnu par des pays tels que la Finlande, la Russie, la Suède, l'Allemagne et la Norvège.

Différentes approches ont été examinées pour la réalisation de ce simulateur de navigation dans les glaces. Il existe au Canada un certain nombre d'installations où des moteurs de simulation modifiables de grande capacité pourraient être utilisés à cette fin. Toutefois, ces systèmes utilisent des technologies brevetées et ne sont pas considérés comme des systèmes «ouverts». En plus d'être très onéreux, ils ne peuvent être distribués facilement ni ne sont facilement disponibles. Avec les ordinateurs perfectionnés actuels, qui offrent une grande performance à un faible coût, vu l'explosion multimédia et la disponibilité d'équipements à réalité virtuelle, il est maintenant possible d'envisager des solutions de rechange aux onéreux systèmes brevetés utilisés actuellement comme simulateurs de navigation. Une nouvelle approche a donc été envisagée pour réaliser un simulateur peu onéreux.

Ce simulateur comporte les fonctions suivantes :

- Simulation des glaces sur le radar de bord
- Simulation et gestion des données de télédétection
- Simulation de l'aspect des glaces durant le jour et durant la nuit
- Simulation de passages de navires (élémentaire à l'heure actuelle)
- Aide d'apprentissage en matière de reconnaissance des glaces et de climatologie des glaces
- Aide d'apprentissage des conditions glacielles autorisant l'entrée dans des régimes de glaces
- Soutien aux systèmes de navigation dans les glaces et au Système électronique de visualisation des cartes marines (SEVCM)

Une partie de cette installation de simulation nécessiterait un module de simulation de radar de bord (MSRB) pour fournir les données de formation des opérateurs de navigation dans les glaces. L'une des opérations critiques de la navigation dans les glaces est l'utilisation d'un radar de marine. Par conséquent, la simulation d'un radar de marine est importante et nécessaire pour tirer le maximum d'avantages du simulateur.

Les données d'un radar à ouverture synthétique (ROS), telles que celles d'un ROS aéroporté ou celles qui sont transmises par un satellite tel que RADARSAT, donnent une image très précise des glaces qui entourent un navire. Quand les images ROS sont collectées, la configuration spatiale et les particularités dans les images sont nettes et elles peuvent être facilement reconnues. Le plus souvent, les particularités détectées par le radar de bord du navire correspondent aux particularités des images ROS. En fait, la correspondance est si grande qu'elle sert de base à la localisation des navires par rapport aux particularités correspondantes dans les images ROS. Les données ROS ont été utilisées comme base de scénario dans le simulateur. Ce projet a donc consisté à utiliser les données ROS avec une modélisation de la simulation radar pour mettre au point la composante de simulation radar du simulateur de navigation dans les glaces.

L'objectif global du programme était de mettre au point une plate-forme de simulation de navigation dans les glaces sur ordinateur personnel peu onéreuse pour former les navigateurs débutants. L'objectif particulier était d'élaborer un module de simulation de radar de bord courant utilisable avec la plate-forme de simulation principale (PSP).

Le MSRB obtient ses données ROS de la PSP et transmet des images radar simulées à ses clients. Il utilise des convolutions d'images de données ROS pour la simulation, ainsi qu'une interprétation de données hors ligne.

Le logiciel de simulation du radar comporte les fonctions clés suivantes :

- brouillage de la distance et de l'azimut
- réduction de la distance
- production d'ombres
- réglage de la portée
- fonctionnement en réseau
- comptabilité avec les logiciels d'interface radar commerciaux actuels

La PSP transmet sur demande au MSRB les données sur la zone examinée (les données rastrées ROS et les données de la table de bits de hauteur). Le MSRB en donne une représentation en coordonnées polaires sur laquelle il produit un flou de distance et d'azimut basé sur l'angle du faisceau et la durée des impulsions. Au besoin, une réduction de la portée est effectuée par les réglages du radar. Finalement, les ombres radar sont produites en temps réel pour créer un masque d'ombre qui est intégré à l'image radar. Les données sont alors mises à la disposition des clients du MSRB.

Cette phase du projet a été un succès malgré certains problèmes techniques au départ avec la performance et la vitesse du système. Le MSRB est opérationnel, mais il reste encore beaucoup à faire. Entre autres, il faut :

- ajouter la trajectoire du navire à l'image radar simulée
- effectuer une interpolation de pixels aux portées inférieures
- ajouter la synthèse d'images à polarisations croisées
- améliorer la simulation des foyers des échos produits par les vagues en mer libre
- ajouter un afficheur d'images autonome
- ajouter un dispositif de réglage en ligne de la simulation
- améliorer la production de la table de bits de hauteur des ombres radar

## TABLE OF CONTENTS


1. INTRODUCTION ..... 1

1.1. SCOPE ..... 1

1.2. OVERVIEW ..... 1

1.3. DOCUMENT OVERVIEW ..... 1

2. DESCRIPTION OF SYSTEM ..... 2

3. BASIS OF RADAR SIMULATION ..... 4

4. MODELLING CONTEXT ..... 5

4.1. MARINE RADAR DISPLAY FEATURES ..... 7

4.2. SAR IMAGE TRANSFORMATION ..... 8

4.3. CONVERSION OF SAR DATA TO MARINE RADAR FORMAT ..... 10

4.4. SIMULATING RADAR RETURNS FROM ADDITIONAL FEATURES ..... 14

4.4.1. RADAR SEA CLUTTER ..... 14

4.4.2. SHADOWS ..... 16

4.4.3. FEATURE RANGE AND AZIMUTH BLURRING ..... 20

5. SPECIFICATION OF SRSM ..... 21

6. SRSM SOFTWARE ..... 23

6.1. SRSM SOFTWARE FUNCTIONAL PROCESSES ..... 23

6.2. SRSM SOFTWARE DATA AND CONTROL FLOW ..... 24

6.3. SRSM SOFTWARE OBJECTS LIST ..... 25

6.4. SRSM PROGRAM FILE LIST ..... 27

6.5. SRSM OPERATING SYSTEM ..... 28

6.6. SRSM SOFTWARE IMPLEMENTATION LANGUAGE ..... 28

7. SYSTEMS LIMITS ..... 29

8. RECOMMENDATIONS AND PROPOSED IMPROVEMENTS ..... 30

9. CONCLUSION ..... 31

APPENDIX A - TECHNICAL AND USER MANUAL

APPENDIX B - SAMPLE OUTPUT IMAGERY

## LIST OF TABLES

Table 4.2-1 Typical PRF and Pulse Lengths for Marine Radar ..... 9
Table 4.2-2 Number of Pulses per Beamwidth for Different PRFs ..... 10
Table 4.3-1 Resolution Table for a One-degree Beamwidth ..... 12
Table 6.3-1 SRM Software Components ..... 25
Table 6.4-1 SRSM Source Code Files ..... 27

## LIST OF FIGURES

Figure 2-1 SRSM Context Diagram ..... 2
Figure 4-1 Radar Backscatter Coefficient versus Incidence (Grazing) Angle ..... 6
Figure 4.2-1 Mapping Polar Data onto a Cartesian Grid ..... 8
Figure 4.3-1 Range and Azimuth Blurring Filter ..... 12
Figure 4.4.1-1 Rayleigh Distribution ..... 15
Figure 4.4.1-2 Negative Exponential Distribution ..... 15
Figure 4.4.2-1 Radial Samples (Height Polygon Edges) ..... 17
Figure 4.4.2-2 Cross Section 1 ..... 17
Figure 4.4.2-3 Cross Section 2 ..... 17
Figure 4.4.2-4 Radial Height Samples ..... 19
Figure 4.4.2-5 Radial Shadow Example 1 ..... 19
Figure 4.4.2-6 Radial Shadow Example 2 ..... 20
Figure 6.1-1 SRSM Functional Block Diagram ..... 23
Figure 6.2-1 The SRSM Data and Control Flow Diagram ..... 24
Figure 6.3-1 Object Diagram ..... 26

# 1. Introduction

## 1.1. Scope

This document contains the final report for the Shipboard Radar Simulation Module (SRSM) Project.

## 1.2. Overview

The use of marine radar is one of the critical elements in ice navigation. This has been proven and documented by actual Arctic field trials and projects sponsored by the Transportation Development Centre (TDC) in the past. These include the Marine-Vue, G2, and IceNav to MRI integration projects. The SRSM is part of the Main Simulator Platform (MSP) and is being developed by TDC and PhiloSoft in order that a full and complete training platform can be developed for ice navigation.

The overall program objective is to develop a low-cost, PC-based Ice Navigation Simulation platform to train entry-level ice navigators. The SRSM application will provide radar images to client applications that request it. It will provide data to these systems by processing SAR and feature data received from the MSP. This project deals specifically with the design and implementation of the SRSM software.

## 1.3. Document Overview

This document contains a brief description of the SRSM's general architecture and functionality. Also included are samples of the imagery generated by it for various range settings and modes. The imagery contains samples of imagery for varying ice types and feature heights. A conclusion, recommendation and proposed improvements summarize the progress made and the additional work needed to be done in order to improve the result. This document also provides several "views" of the SRSM software. These include:

- a context diagram and description,
- a functional process diagram,
- a data and control flow diagram,
- a source code file list.

Appendix A contains the Technical and User Manual, while Appendix B contains sample output imagery.

## 2. Description of System

In this project the MSP provides the data to and interfaces with the Shipboard Radar Simulation Module (SRSM). The SRSM in combination with the Ice Navigation System Radar Display and/or the MRI Display Software permits the operator to view synthesized marine radar imagery.

Synthetic Aperture Radar (SAR) data, such as that taken by airborne SAR or even from a satellite such as RADARSAT, represents a very accurate picture of a ship's ice environment. When the SAR imagery is collected, the spatial geometry and features in the imagery are accurate and the features are readily identifiable. In most cases, features visible on the shipboard radar will correspond to equivalent features in the SAR imagery. In fact, the correlation is so strong that it forms the basis for locating a vessel relative to equivalent features in the SAR imagery. The SRSM will perform the function of converting the SAR data and the associated feature file data into imagery that can be used to simulate marine radar.

Figure 2-1 illustrates the context of the SRSM in the MSP system. It shows the general relationship of clients and servers between the MSP and the SRSM.

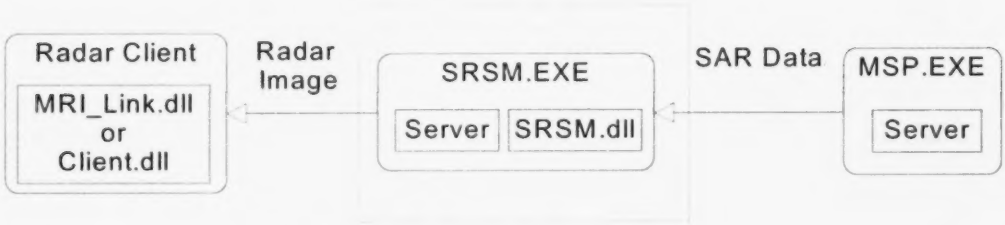



**Figure 2-1 SRSM Context Diagram**

The SRSM is a server application that provides simulated marine radar imagery to client applications that request it. It does so by querying the MSP for a block of SAR data. The data provided by the MSP is comprised of the original SAR data plus a height bitmap of the corresponding area. The height bitmaps are generated with a separate tool during the initial SAR data analysis phase and are created off-line. This is because of the sheer volume and the number of operations needed to be performed when the height bitmaps are created.

Once the SRSM gets its data from the MSP, it converts the data from X-Y Cartesian coordinates into R-θ Polar Coordinates. All the range attenuation, range and azimuth blurring and radar shadow generation are performed in the R-θ domain.

**General**

The SRSM software is a set of Dynamic Link Libraries (DLL) and executable programs (EXE) that will run in conjunction with the MSP software. The purpose of the SRSM is to provide a module that will simulate marine radar imagery, based on the analysis of Airborne SAR imagery, to radar clients. The SRSM is used in conjunction with radar client display applications to assist the trainee in assessing ice conditions as viewed in a marine radar context. The SRSM

supplies an interface to radar clients that display and control the SRSM. The SRSM gets data from the MSP.

The SRSM software provides data to radar clients by first requesting SAR data and feature information from the Main Simulator Platform (MSP). It then processes this data by adding range and azimuth blurring, range attenuation, STC and FTC processing, sea clutter where there is not SAR data and finally feature shadowing. Once data is processed, it hands that data back to the requester. The radar client also has the capability to control the SRSM by adjusting the range setting and radar view (North Up, Ships' Head Up) as well as selecting the type of processing that will be applied to the SAR image data. This will directly affect the type of data requests that are made to the MSP and the processing that the SRSM must perform. Whenever a radar client requests data, whether it is ship position, heading or image data, it performs a call to the client dynamic link library (labeled Dll_link.dll).

The SRSM.exe takes this call and hands over the latest updated information to the radar client. The SRSM.exe has a separate thread that requests regular updates from the host MSP and processes this information on a timed interval basis, independently of the requests made by the radar client. The SRSM software therefore is designed as a multi-threaded system running under Windows NT 4.0 with four threads or run-time tasks that make up the system. These include:

- SRSMApp,
- MSPClient,
- RadarDataProcessor,
- SRSMServer Threads.

These threads will be run in a time sliced fashion, as are all MS Windows tasks.

## 3. Basis of Radar Simulation

This section contains modelling theory and deals with the SAR to marine radar data mapping in two steps. First, it defines the relationship between the synthetic aperture radar and the equivalent features found by a shipboard radar including:

- the mapping of SAR data to Marine Radar data,
- the effects of radial/range blurring,
- the effects of incidence angle relationship between SAR data and radar image,
- the effects of range attenuation,
- the effects of Sensitivity Time Control on range attenuation,
- the effects of Fast Time Constant on edge detection.

Second, it further illustrates the relationship between radar detectable features such as the position and size of icebergs, ridge heights, etc. and those same features that are not readily available in the SAR data but are available by other means. These include:

- Position and Extents of Icebergs,
- Land Outline,
- Ridges,
- Open Water and Ice.

# 4. Modelling Context

This project required the development of techniques for the simulation of a marine radar image that should appear on a radar display given a particular ice situation. Synthetic Aperture Radar (SAR) will be used as the basis for the simulation. Two basic techniques will be used. The first will provide transformations to be applied to the SAR image so that it exhibits the same characteristics as a marine radar signal. The second will provide for the introduction of specific features into the radar signal that has been identified from other sources.

Radar backscatter from icebergs and sea ice is a complex phenomenon dependent upon the angle of incidence and ice properties such as conductivity, surface roughness and ice type. At microwave frequencies, conductivity plays a major role with the better reflection coming from the higher conductivity media. In general, this is the reason that icebergs, composed of fresh water, are poor radar reflectors. Similarly, multi-year sea ice will have a lower conductivity than first-year ice as a result of successive seasons of brine drainage during the summer months. It may not be necessary to fully model the complex behavior of radar backscatter from ice for this simulation as the SAR image provides an actual measurement of the radar backscatter for the particular case under consideration. The major difference between the SAR image and the required marine radar image is the incidence angle. Airborne SAR may have incidence angles in the range of 10 to 50 degrees while for marine radar the incidence angles will range from 0.1 to 5 degrees. Furthermore, the SAR processor may use techniques to normalize the radar signal so that the images have no range dependence.

Herein lies a problem. The incident angle dependence is not linear over the range of angles under consideration. In the higher incidence angle region typical of SAR, the backscatter coefficient may exhibit a linear fall off, however, as more shallow angles are approached, the fall off may become more dramatic. Figure 4.1 illustrates the type of angle dependence that radar backscatter may exhibit. This angle dependence affects the range dependence of the received radar signal in addition to the normal spherical spreading loss, resulting from an $R^{-4}$ signal fall off. The basic relationship is defined by the radar equation and is given in Equation 1.

$$P_r = (P_t G^2 \lambda^2 \sigma)/((4\pi)^3 L R^4) \qquad (1)$$

Where
- $P_r$ = Power received (watts)
- $P_t$ = Power transmitted (watts)
- $G$ = Antenna gain
- $\lambda$ = Radar wavelength (m)
- $\sigma$ = Target radar cross section ($m^2$)
- $L$ = Radar losses
- $R$ = Range to target (m)

The radar cross section, $\sigma$, for a point target is independent of the radar resolution. A point target is defined as being smaller than the radar resolution cell size. For distributed targets such as sea clutter and ice, the radar cross section may be described by Equation 2.

$$\sigma = [(R\,\theta)\,(c\,\tau)/2]\,\sigma^{\circ} \qquad (2)$$

Where $\sigma^{\circ}$= Radar backscatter coefficient ($m^2/m^2$)
$\theta$ = Antenna horizontal beamwidth (radians)
R = Range to target (m)
c = Speed of light = $3 \times 10^8$ (m/s)
$\tau$ = Radar pulse length (s)

By substituting Equation 2 into Equation 1 it may be seen that the received signal from the distributed target will have an $R^{-3}$ range dependence coupled with the incidence angle dependence demonstrated in Figure 4.1 (Equation 3).

$$P_r = [(P_t\,G^2\,\lambda^2\,\theta\,(c\,\tau)/2)/((4\pi)^3\,L\,R^3)]\,\sigma^{\circ} \qquad (3)$$

If it is assumed that the backscatter coefficient is defined by Figure 4.1, it is possible to

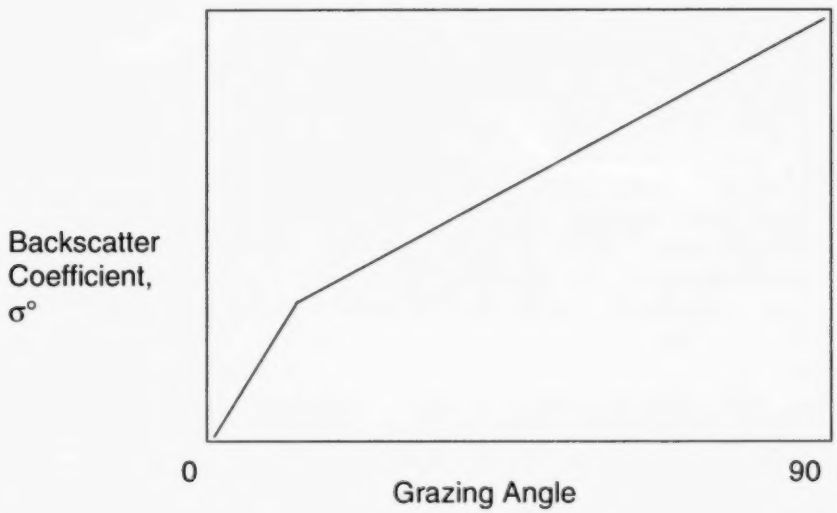



**Figure 4-1 Radar Backscatter Coefficient versus Incidence (Grazing) Angle**

define the effect of ice ridges as a tilting of the ice towards the radar and thereby changing the incidence angle. A vertical ridge face would exhibit a larger $\sigma^{\circ}$ than flat sea ice. The situation is complicated by the fact that $\sigma^{\circ}$ for first-year ice, multi-year ice and icebergs will differ.

In summary, it is likely that the SAR image has been range normalized so that the only information remaining is the relative intensity of the radar return from the various ice features. We must therefore describe a mapping of these intensities to those that would be observed for the marine radar case. This could be simply accomplished by choosing a model for $\sigma^{\circ}$ and applying it to Equation 3. The most simplistic model could have $\sigma^{\circ}$ behave as $R^{-1}$ or $R^{-2}$ for all incidence angles resulting in an overall range dependence in Equation 3 of $R^{-4}$ or $R^{-5}$.

## 4.1. Marine Radar Display Features

The following section presents the basic factors that control the appearance of a radar signal on a radar display. A typical raster scan marine radar display presents the radar signal return in either color or monochrome with typical amplitude resolution of two bits representing the signal in four shades or colors. Older analogue displays utilized a long persistence phosphor and could provide a dynamic range of in the order of 4 bits or 16 shades of intensity. In the last 10 years new radar display technologies have emerged that can yield even more resolution. The Marine-Vue Radar Display system supports a true 8 bits and 256 shades of green. The SRIC (Shipboard Radar Interface Component) together with the MRI (Modular Radar Interface) and even the Titan Radar all produce true 8 bit resolution in the digital domain. This permits digital signal processing to be applied to the digitized radar signal so as reduce the noise artifacts of the radar signal.

The processing normally available with marine radars include FTC (Fast Time Constant) and STC (Sensitivity Time Control). The FTC control is a high pass filter and was designed for the removal of noise like clutter typical of rain and is therefore often referred to as the Rain Clutter control. The STC control is used to remove the range dependence in the received radar signal and thereby normalize the radar signal with range. The use of these controls with the low dynamic range display tends to give acceptable performance for point target detection but results in poor definition of distributed targets such as continuous ice cover. When navigating in ice these displays may be adjusted to remove the majority of the sea ice signal leaving significant features such as ice ridges and icebergs.

The physical and radar processes controlling how a radar signal appears on the radar display include:

1. **Radar signal range dependence.** The received signal from a point target decreases in amplitude proportional to $R^{-4}$. As discussed in the previous section for distributed targets such as sea clutter and continuous ice cover, the signal will decrease in amplitude proportional to $R^{-4}$ to $R^{-5}$ and will depend upon the model chosen for the backscatter coefficient. If STC or FTC is not used the radar signal close to the center of the radar display will be much brighter than the signals from longer ranges.

2. **Radar pulse length.** The radar pulse length governs the minimum feature that can be resolved in the radial direction from the radar. This is referred to as the range resolution of the radar.

3. **Radar beamwidth.** The radar beamwidth governs the minimum feature that can be resolved in azimuth. For marine radar, the azimuth resolution is defined as the beamwidth times the range, and will therefore degrade with increasing range.

4. **STC control.** The STC control is often referred to as the Sea Clutter control and may vary in its implementation. When the control is designed to normalize the sea clutter the STC curve provides a time varying gain designed to remove the range dependence.

5. **FTC control.** The FTC control is implemented as a high pass filter. As the control is adjusted the time constant of the filter is altered. The shorter the time constant the more the output of the filter will follow the input.

6. **Analog to digital conversion.** The FTC, STC and GAIN controls are normally applied to the analog signal. The processing is normally done in 8 bits.

## 4.2. SAR Image Transformation

The following section describes how a SAR image may be transformed to simulate the marine radar image. The inputs to the MRSM will be a geo-referenced SAR image with a separate data file of geo-referenced features. The following review is directed at dealing with the SAR image to marine radar image transformation.

The SAR image will have the following attributes:

Pixel dimension: units (x and y)
Pixel position: Latitude and Longitude or x and y offset from a Latitude and Longitude
Pixel Intensity: Processed radar return (range normalized)

The basic pixel dimension will be governed by the radar resolution. The radar data from a SAR image is normally processed such that the pixels have square geometry. In real aperture radar, such as marine radar, the range resolution is governed entirely by the transmitted pulse length. The azimuth (or across track for SLAR) resolution is a function of the radar beamwidth and is governed by the formula:

Azimuth Resolution (m) = Range (m) x Antenna Beamwidth (radians)

The marine radar has a real aperture and therefore the azimuth resolution degrades with increasing range from the radar. For example, typical marine radar may have a horizontal beamwidth of 1 degree (or 0.01745 radians) so at a range of 100 meters the azimuth resolution is 1.745 meters, at 1000 meters the azimuth resolution is 17.45 meters and at 5000 meters the azimuth resolution is 87.3 meters.

Figure 4.2-1 illustrates the mapping of a real aperture onto a Cartesian grid. The radar beamwidth illuminates more Cartesian pixels with increasing range. The range resolution remains constant with range.

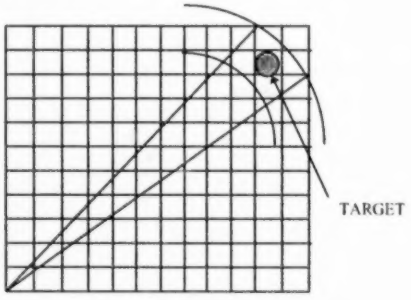



**Figure 4.2-1 Mapping Polar Data onto a Cartesian Grid**

It is desired to generate marine radar data from Cartesian format SAR data. The SAR data does not contain the distortion of the radar echo caused by the change in azimuth resolution with range as seen in marine radar. In order to generate a realistic marine radar image this effect should be simulated. If the SAR data is left in Cartesian coordinates this distortion will not be easily achieved, as the distortion is azimuth and range dependent, as can be seen from

Figure 4.2-1. If the SAR data is converted to polar format with the desired ship position as the origin then this beamwidth distortion may be implemented using a filtering process with an azimuth filter having the characteristic of an integrator. The details of the inverse scan conversion process may be worked out when the desired input and output resolutions are known. There are other benefits to working in polar format as this is the natural format of the marine radar data. The implementation of FTC (Fast Time Constant) and STC (Sensitivity Time Control) are easier and more effective in polar format as the processes operate in the range direction only.

As the marine radar antenna scans through 360 degrees it illuminates targets by generating radar pulses at the pulse repetition frequency (PRF). The PRF is normally linked to the radar pulse length with the longer radar pulses having the lower PRF. Table 4.2-1 presents typical radar pulse lengths, corresponding PRF values and the radar resolution for the particular pulse length. The radar range resolution is given by:

Range Resolution (m)= (3E8) x (Pulse Length(s))/2

**Table 4.2-1 Typical PRF and Pulse Lengths for Marine Radar**

| PRF(Hz) | Pulse Length(μs) | Range Resolution(m) |
|---|---|---|
| 800 | 1.00 | 150.0 |
| 1600 | 0.25 | 37.5 |
| 3200 | 0.05 | 7.5 |

A point target (or a target smaller that the azimuth resolution of the radar) will be hit by the number of radar pulses in the radar beamwidth. The number of pulses per beamwidth, N, may be calculated from the following expression:

N = (Beamwidth (degrees) x PRF (Hz))/(6 x Antenna Speed (RPM))

Table 4.2-2 presents N for Marine Radar having a 1-degree beamwidth operating at 24 RPM antenna.

**Table 4.2-2 Number of Pulses per Beamwidth for Different PRFs**

| PRF(Hz) | N(pulses) |
|---|---|
| 800 | 5.5 |
| 1600 | 11.1 |
| 3200 | 22.2 |

From Table 4.2-1 and Table 4.2-2 it may be seen that for standard marine radar the number of pulses per beamwidth is highest for the shortest pulse (highest PRF). In the case of ice navigation it is expected that most ice navigation tasks will utilize the two shorter pulses having range resolutions of 37.5 and 7.5 meters respectively. In the case of SAR, the range and azimuth resolution will depend upon the radar's operating mode. It is possible that the following different situations will occur with regards to range resolution:

- SAR range resolution better than radar range resolution. This would typically happen when the radar is operating on long pulse. This is the preferred situation as it is then possible to degrade the SAR data to match the resolution of the marine radar.
- SAR range resolution equal or close to the radar range resolution. This would probably happen when the radar is operating on medium pulse. In this case no processing of the data in range will be necessary.
- SAR range resolution worse than the radar range resolution. This will occur when the radar is operated on short pulse. This is the worst case scenario as it will not be possible to extract the full desired radar range resolution from the SAR data.

As the source of SAR data may vary it is expected that the resolution of the data will vary as well and a method for providing the best match of radar resolution to SAR resolution should be included.

## 4.3. Conversion of SAR Data to Marine Radar Format

In brief, the following steps would see the generation of a marine radar image from a SAR image. Note that this procedure converts the SAR data to polar format for processing. A less effective approach would operate on the Cartesian format data.

1. Define the radar parameters (if any) that will be fixed for the simulator. This may include the radar beamwidth and antenna scan rate.
2. Define the radar image resolution in Cartesian coordinates. For example, the desired image size might be 1024 x 1024 pixels.
3. Determine the SAR image resolution.

4. Select the radar operating mode:
    - Range and pulse length
    - STC processing
    - FTC processing
    - Other filters, etc.

5. Determine the ship position from the simulator and from the radar range define the area of interest of the SAR image for the SRSM's local buffer.

6. Using the desired output resolution and input resolution compute a look-up-table for scan conversion. If the input resolution is a 1024 x 1024 pixel segment of a SAR image and it is desired to generate 1024 radials (or pulses) x 512 range samples a look-up-table of 4 MB size would be sufficient. If the data were processed by quadrant then a 1 MB LUT would be adequate. It would be beneficial to standardize the output resolution this way, as the LUT approach to scan conversion is quite efficient.

7. Once the polar data is available, a two-dimensional filter may be run over the data to introduce the desired degradation and processing effects. Alternatively, LUT tables based on range setting applied to the original Cartesian SAR data might produce a better result. This will be considered during the design and prototyping stage.

The filter will have two components, a range component and an azimuth component. The algorithm will compute the average of the pixels addressed by the filter. If the desired marine radar range resolution matches the SAR resolution then the filter has dimension 1 in the range direction. If the desired marine radar range resolution is worse than the SAR resolution then the filter has dimension that is the ratio of (Marine Radar resolution /SAR resolution) in the range direction.

For example, if the SAR has a resolution of 25 meters and the radar has a resolution of 150 meters then the filter should average 6 pixels in range to obtain the desired marine radar resolution.

The azimuth filtering will depend only upon the beamwidth. 1024 azimuths will provide about 2.8 samples per beam width for a 1 degree antenna beamwidth. The range setting will affect the number of adjacent pixels to be averaged for a given resolution. At 3 nm for example, the azimuth resolution for a 1 degree beamwidth is ~ 100 meters. Therefore 4 adjacent SAR pixels (@ 25 M/ pixel) need to be averaged in order to recreate the azimuth blurring effect. At 6 nm, 8 pixels of the same resolution need to be averaged. For range settings above 6 nm, the pixel resolution of the SAR will drop to between 50 and 200 m per pixel. This is in order to keep network transfer reasonable, say under 2 seconds. Table 4.3-1 shows the relationship between the pixel resolution being delivered from the MSP and the respective range and azimuth resolutions based on range setting (i.e. on long, medium or short pulses).

**Table 4.3-1 Resolution Table for a One-degree Beamwidth**

| | SAR Image Resolution from MSP | | Range Resolution | | Azimuth Resolution | | Degree Arc |
|---|---|---|---|---|---|---|---|
| Range NM | # Pixels | # Meters | # Pixels | # Meters | # Pixels | # Meters | # Meters (Approx.) |
| 48.0 | 1 | 200 | .75 | 150 | 8 | 200 | 1600 |
| 24.0 | 1 | 100 | 1.5 | 150 | 8 | 100 | 800 |
| 12.0 | 1 | 50 | .75 | 37.5 | 8 | 50 | 400 |
| 6.0 | 1 | 25 | 1.5 | 37.5 | 8 | 25 | 200 |
| 3.0 | 1 | 25 | 1 | 25 | 4 | 25 | 100 |
| 1.5 | 1 | 25 | 1 | 25 | 2 | 25 | 50 |
| .75 | 1 | 25 | 1 | 25 | 1 | 25 | 25 |

It is expected that in the worst case the filter size would be no greater than 8 x 1.5. So for every output pixel (in this case 1024 x 1024) 12 adds and one divide (or shift) will be required, resulting in about 13 million operations per scan for the range azimuth filter. In actuality only half of these operations will take place as on any given radial there would be no azimuth blurring at the origin and the maximum would occur at the end of the radial. Therefore this number reduces to approximately 6.5 million operations. On a 200 MHz Pentium Pro this processing load may not be significant.

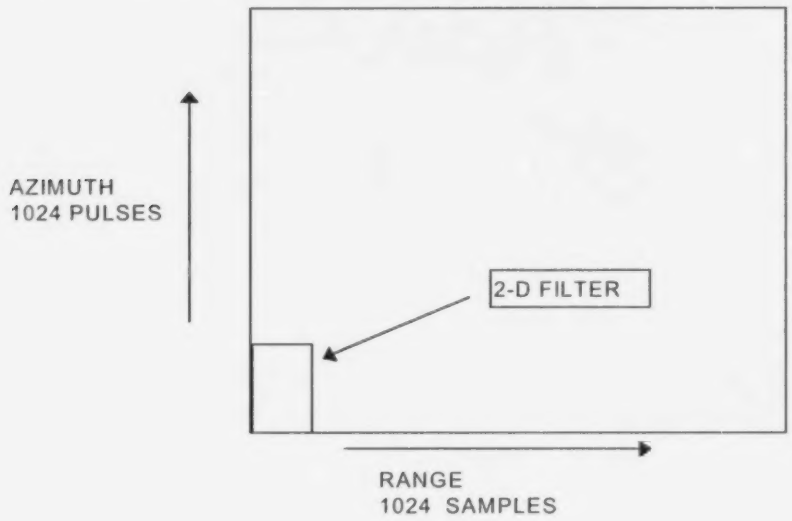



**Figure 4.3-1 Range and Azimuth Blurring Filter**

8. Apply range dependent algorithms such as FTC and STC.

The FTC algorithm will be a high pass digital filter operating over a span of range samples. The desired effect could be achieved by forming the average of a predetermined number of range samples and subtracting the result from the sample being processed.

The STC algorithm is required to simulate the effect of range dependence in the radar data. SAR data is normally corrected to remove range dependence thereby providing an equal intensity return as a function of range. The SAR image already has the STC applied. In order to simulate the marine radar signal the STC will have to be removed and then reapplied based on typical implementations.

The received radar signal strength from marine radar has range dependence that is inversely proportional to the range to the fourth power. As discussed earlier, for distributed targets the backscatter coefficient may alter this range dependence. Most marine radars use a logarithmic amplifier so that the voltage out of the receiver is the logarithm of the received signal strength. Therefore, signal intensity (E) may be expressed approximately by the formulae below. Initially, the radar return signal strength will be of the order:

$$E = K1/R^4$$

This signal is usually fed into a logarithmic amplifier. For a given E, the output of the amplifier will look like:

$$LOG\ (E) = LOG\ K1 - LOG\ (R^4)$$

Arithmetic manipulation yields:

$$LOG\ (E) = K2 - 4*LOG\ (R)$$

Applying gain to the logarithmic output creates:

$$K3\ (LOG\ (E)) = V = K3*\ (K2 - 4*\ LOG(R))$$

And therefore:

$$V = K4 - K3*4*LOG(R)$$

The net result of the above yields: Intensity K4 - 4*K3* LOG (R)

So if the maximum pixel intensity (K4) is 255 at zero range, and K3 (an intensity-scaling factor) is 20 then at a range of 10000 meters the maximum intensity will be:

Intensity 255 - 4*(K3)* LOG (10000) = 255-180= 65

The exact value of K3 will be determined by experimentation with existing data sets such as radar samples from the data acquired from the Modular Radar Interface Field Trials. Also results that yield a negative value for intensity will be treated as Intensity equaling zero.

This defines the mapping of pixel intensities to remove the STC applied in the SAR image. In order to apply STC the algorithm should reverse the above process in stages dependent upon the amount of STC applied.

## 4.4. Simulating Radar Returns from Additional Features

Besides the SAR data, the SRSM will also use a secondary set of associated feature data as a basis for recreating realistic marine radar images. This data will consist of a set of feature outlines complete with context information attached to it and will be provided to the SRSM by the MSP. This data will be generated from an off-line detailed analysis of feature height and extent. Attributes of features that will affect the SRSM simulation will include size, shape and surface condition. The height extent of a feature will cause shadowing of the region behind it. The length of the shadow will depend upon the feature height and ship antenna height. The position, extents and height information will be used to help the SRSM to create the correct shadow size based on radar incidence angle. Only position, height and extent information tell the SRSM where objects are. This will apply for both open water and ice-covered areas. By using this height model, the SRSM will be able to handle any object, be it stationary as in the case of land, or moving as in the case of a moving ship.

A typical object or feature would have a length, width, orientation and height. Figure 4.2-1 illustrates the basic geometry of the radar resolution cell when mapped to Cartesian space. It may be seen from the figure that the orientation and size of a point target echo is often related to the target position and not its physical size and orientation. This means that the orientation of the point target will change as the radar (vessel) moves by it requiring that the simulator be able to change the orientation of the echo on an update-by-update basis. This may be best accomplished by handling these features in polar format. The polar to Cartesian conversion process will then take care of the orientation issues. For targets that have large extent, such as large icebergs and ridges, the orientation and size will depend more upon the physical size and orientation of the feature.

### 4.4.1. Radar Sea Clutter

In simulation scenarios where there is no SAR data available (i.e. empty SAR files), the SRSM will be capable of synthesizing the open water radar sea clutter data. Simple radar sea clutter models are available that calculate $\sigma^\circ$ required for Equation 2 from ocean weight, wind speed and direction. These models provide average $\sigma^\circ$ and may be used to set the background clutter level. The sea clutter will also have a fluctuation statistic that will vary from Rayleigh distributed to K-distributed for high-resolution modes at shallow incidence angles. For the purposes of this simulation it is expected that the Rayleigh distributed clutter will be satisfactory.

The voltage output (V) of the radar receiver will be given by the Rayleigh distribution as is defined as:

$$F(V) = (V/S^2)e^{-V2/2S^2} \text{ where } S^2 \text{ is the variance.}$$

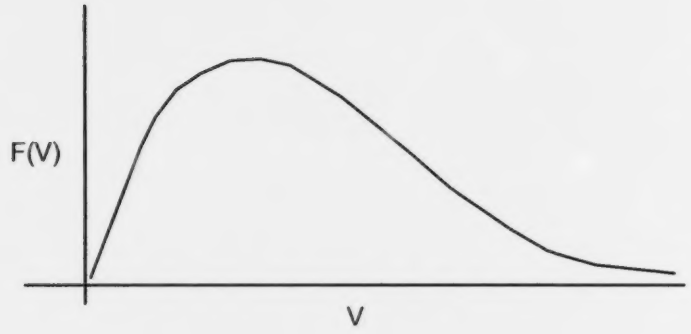



**Figure 4.4.1-1 Rayleigh Distribution**

And the receiver output power (P) will be given by the Negative Exponential distribution defined as:

$$F(P) = 1/P_{avg}e^{-P/Pavg}$$ Where: $P_{avg}$ is average power.

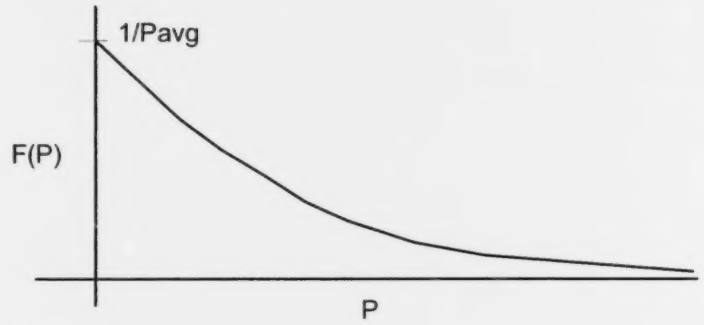



**Figure 4.4.1-2 Negative Exponential Distribution**

The clutter distribution will be scaled by the mean power. The method for computing the pixel value, given the mean signal power, will be generated by shaping the output distribution to look like the Negative Exponential Distribution defined above.

Finally, the pixel value will be proportional to the log of the received power when a log amplifier is used so the last step in computing the pixel value is the computing of the log of the power value.

The simulation of sea clutter is achieved by creating random pixel intensities in the polar domain. This polar data can then be overlaid with obstacle or feature data as described in the sections below.

In cases where there are icebergs in open water, synthesized SAR iceberg bitmaps will be generated off-line during SAR image analysis and passed over to the SRSM at run-time. The

ice feature bitmaps will need to be synthesized in order that the correct simulation of the edge and shadowing can be performed.

### 4.4.2. Shadows

The radar shadow is best defined as a pixel mask that will be applied to the marine radar image. The mask will be calculated from the position and height of features in the radar field of view. As the orientation of the shadows will change for each new vessel position, it will be necessary to recompute the mask for each new vessel position. If the pixel mask is initialized as all ones and the shadows are drawn onto it, then a bitmap *and* operation will add the shadows to the marine radar image. Alternatively, the shadows may be directly drawn onto the image data. Radar shadows should be calculated in the polar coordinate system as this facilitates easy computation and can easily be applied as a mask on the corresponding polar scanned data.

The shadows will be generated based on detailed height analysis of the original SAR data. In order to generate a more realistic radar image some smoothing of the shadows and target detections will be required. This smoothing is typical of real world marine radar. The radar Simulator would require a height map that would coexist with the SAR data. This height map will contain height information for each and every pixel in the SAR data. Because this height data is currently not available, it will be necessary to synthesize it. The synthesis will be based on manual interpretation of the ice features in the SAR data as well as any other information such as land and ships. The information can be coded as a set of closed contour polygons with height data. These polygons would represent outlines of features such as ridges as well as icebergs. Additional features such as ships and icebergs can be added in the form of bitmaps.

**Height Polygon Edges**

The edges of the height polygons should be smoothed such that the center of the line used to draw the polygon would have the maximum height and the adjacent polylines would gradually diminish in height when moving away from the line center. In Figure 4.4.2-1 the "O" symbol represents the radar origin and P11 and P12 are radials that intersect the feature polygon H1 at A and B. The perimeters of the ice floes are marked by "veins" of varying thickness. A cross section of the "veins" can be seen in Figures 4.4.2-2 and 4.4.2-3. These "veins" are used to smooth the hard transitions of the polygon edge. The pixels adjacent to the center of the "vein" would have a height lower than that the center of the "vein".

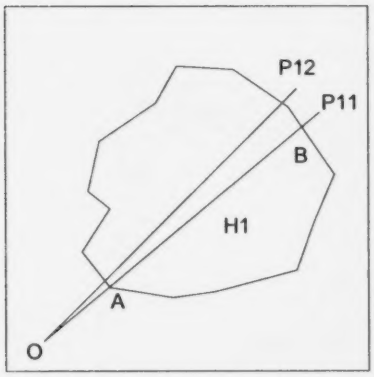



**Figure 4.4.2-1 Radial Samples (Height Polygon Edges)**

**Figure 4.4.2-2 Cross Section 1**

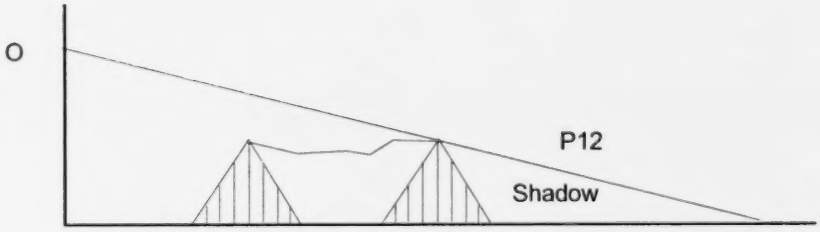



**Figure 4.4.2-3 Cross Section 2**

**Filling the Height Polygons**

The area or region bounded by the polygon would have height information recreated from the pixel values of the actual SAR image itself. For example, let us say that the region mapped out in Figure 4.4.2-1 represents a region whose approximate height equals 1 meter. And let us also say that the pixels in the area have a range from 0 to 255 and that the nominal average of the

pixels in that region is 20. The value twenty would represent 1 meter of height. The pixels that are darker i.e. < 20 nominal would have a lower height while those that are greater would have a greater height. This would in fact coincide with the reality that brighter SAR image pixels generally represent thicker and therefore higher ice. When implemented, such a scheme could provide for the simulation of pixel height for each and every pixel.

**Height Modelling**

**Scheme 1**

One approach is to use the value of the pixels in the SAR data that are bounded by the feature polygon. Using the formula below it is possible to simulate the height value for each pixel in the region.

$$H_{Pixelxy} = XY_{Pixel\ Intensity}/\text{Maximum Pixel Intensity in Region}$$

Where:

$H_{pixelxy}$ is adjusted height of pixel XY inside region

$XY_{Pixel\ Intensity}$ is SAR pixel XY intensity in region

In this approach the perimeter would represent the maximum height with the variations in the bounded plane being less than or equal to the perimeter.

**Scheme 2**

A second approach could involve using the average intensity of the pixels of the bounding region and using that value in the formula below.

$$H_{Pixelxy} = [XY_{Pixel\ Intensity} * Height_{Perimeter}]/\text{Average Pixel Value in Region}$$

Where:

$H_{pixelxy}$ is adjusted height of pixel XY inside region
$XY_{Pixel\ Intensity}$ is SAR pixel XY intensity in region
$Height_{Perimeter}$ is perimeter height of region

In this approach the perimeter would represent the nominal height with the variations in the bounded plane being less than or equal or greater to the perimeter. Some bounding of pixel height will be performed in order that bright pixels do not exceed the perimeter boundary by more than a factor of 2. This scheme presents the preferred approach.

**Scheme 3**

A third scheme simplifies the approach of scheme #2 by averaging only the pixels on the perimeter of the polygon to computer instead all of the pixels. However, this approach makes the selection of the edge more critical. The computation of the height map could be performed by the SRSM, but if CPU resources are too limited then this could in fact be performed off-line, (while the SAR image analysis was done for example), so that the pixel average computation would not burden the SRSM process. The SAR bitmap and height bitmaps could be passed to the SRSM that would then convert them to polar format. The shadows would be "anded" with

the SAR. Fig 4.4.2-4 illustrates two radar radials P21, P22 In the case of P21 we have radial scanning across two flows H2 and H3. The ice floe H2 is lower that the flow H3. In this case the shadows will be generated due to the depression of H2. See fig 4.4.2-5 In the case of P22 there is a progressive increase in height as the radial crosses the H1, H4 and H5 polygons. Shadows will occur as a result of the decline in height after H5. See Fig. 4.4.2-6

**Baseline Height**

The concept of baseline height should be added to the simulation, as the height modelling is purely heuristic in nature. The simulation should allow for the adjustment of this baseline height. In other words, it should be possible to experiment with the ratio of ice height to intensity so that the best radar image results can be achieved.

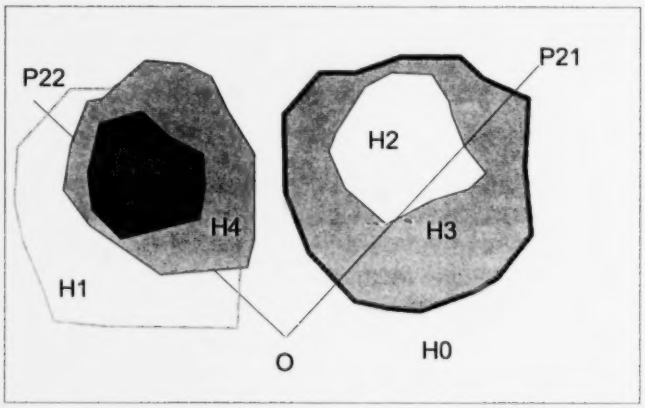



**Figure 4.4.2-4 Radial Height Samples**

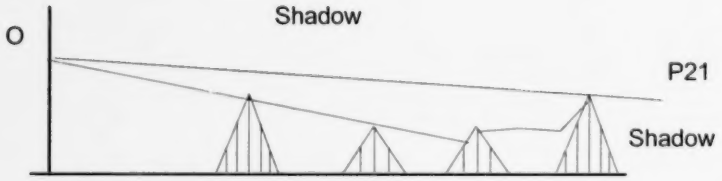



**Figure 4.4.2-5 Radial Shadow Example 1**

the SAR. Fig 4.4.2-4 illustrates two radar radials P21, P22 In the case of P21 we have radial scanning across two flows H2 and H3. The ice floe H2 is lower that the flow H3. In this case the shadows will be generated due to the depression of H2. See fig 4.4.2-5 In the case of P22 there is a progressive increase in height as the radial crosses the H1, H4 and H5 polygons. Shadows will occur as a result of the decline in height after H5. See Fig. 4.4.2-6

**Baseline Height**

The concept of baseline height should be added to the simulation, as the height modelling is purely heuristic in nature. The simulation should allow for the adjustment of this baseline height. In other words, it should be possible to experiment with the ratio of ice height to intensity so that the best radar image results can be achieved.

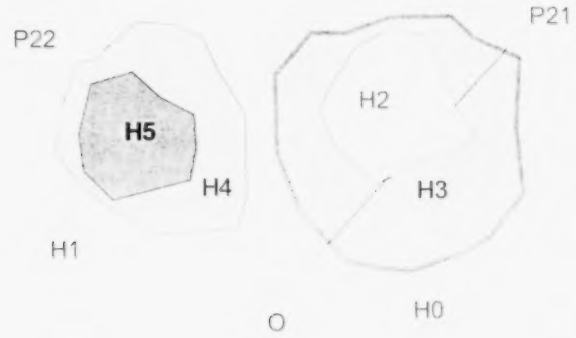



**Figure 4.4.2-4 Radial Height Samples**

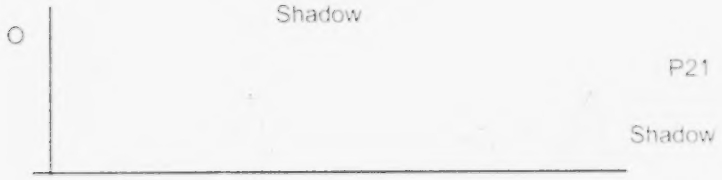



**Figure 4.4.2-5 Radial Shadow Example 1**

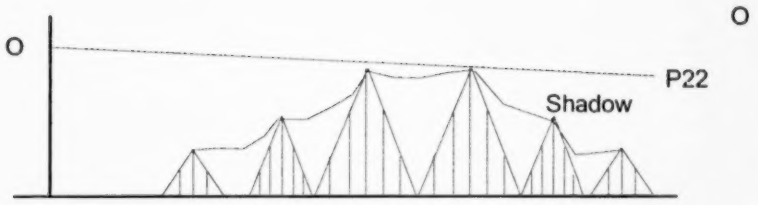



**Figure 4.4.2-6 Radial Shadow Example 2**

### 4.4.3. Feature Range and Azimuth Blurring

The range and azimuth blurring will be applied to both original SAR data as well as the polarized vector data. By doing this the resolution cells will take into account the fact that shadow edges cannot be more "detectable" than the source radar resolution cell. The blurring will be computed once, after the SAR and shadows are "anded" together and just before the conversion back to Cartesian bitmap.

# 5. Specification of SRSM

The SRSM client interface is identical to MRI dynamic library specification. However, not all of the functions are fully simulated. In cases where the function is not simulated, the function call just returns a success code to the calling application.

The SRSM functionality comprises the following components:

- SRSMApp
- Config
- SRSM Data Structures
- Data Management
- Timer Function
- Network Management and Communications
- DLL_link.DLL Or Client.DLL
- SRSM Server
- MSPClient
- SRSM.DLL
- SRSM Main Control
- SAR Image and Feature Processing
- Sea Clutter Generator

The paragraphs that follow describe the software components that address each of these requirement elements.

The list below contains the names the interface calls that are currently simulated by the SRSM.

**Function Names**

CLIENT_STATUS SetTitanOnline(SHORT TargetMRI=-1)

CLIENT_STATUS SetTitanOffline(SHORT TargetMRI=-1)

CLIENT_STATUS SetRadarOnline(SHORT TargetMRI=-1)

CLIENT_STATUS SetRadarOffline(SHORT TargetMRI=-1)

CLIENT_STATUS GetTitanStatus(PTITAN_DEBUG_HDR pStat, SHORT TargetMRI)

CLIENT_STATUS GetProcessor (PSCANFMT pScanFmt, SHORT TargetMRI)

CLIENT_STATUS SetDesiredMode (USHORT Mode, SHORT TargetMRI=-1)

CLIENT_STATUS SetScans (USHORT Scans, SHORT TargetMRI=-1)

CLIENT_STATUS SetPF (SHORT PfPosNeg, SHORT TargetMRI=-1)

CLIENT_STATUS SetCFAR (SHORT CFARPosNeg, SHORT TargetMRI=-1)

CLIENT_STATUS GetData(PSCANFMT header, BYTE *data, SHORT dataType, SHORT TargetMRI)

CLIENT_STATUS GetTitanRanges (PTITAN_MODES pRanges, PUSHORT pModes, SHORT TargetMRI)

CLIENT_STATUS QueryNumOfMRIs (USHORT *NumOfMRIs,char MRINames[4][16])

CLIENT_STATUS GetMRIStatus (MRI_STATUS *status, SHORT TargetMRI)

CLIENT_STATUS SetGyro(F_REAL gyro, SHORT TargetMRI)

The SRSM Server will also supply an interface to set to the STC and FTC.

CLIENT_STATUS SetFTCIndex(SHORT newIndexValue)

**Returned Data**

typedef DWORD **CLIENT_STATUS**;

All DLL functions will return the following values:

CLIENT_STATUS_NO_ERROR

CLIENT_STATUS_INVALID_ARGUMENT

CLIENT_STATUS_FUNCTION_NOT_SUPPORTED

CLIENT_STATUS_NETWORK_ERROR

CLIENT_STATUS_SERVER_BUSY

CLIENT_STATUS_FUNCTION_FAILED

CLIENT_STATUS_QUAD_ERROR

CLIENT_STATUS_MAX_CONNECTIONS

CLIENT_STATUS_ERROR_CONNECTED

CLIENT_STATUS_NO_CONNECTION

CLIENT_STATUS_INVALID_ID

A complete description of these calls can be found in the MRI specification.

# 6. SRSM Software

## 6.1. SRSM Software Functional Processes

Figure 6.1-1 illustrates the main functional processes in the SRSM software. As can be seen from the diagram, the SRSM takes both SAR and feature data from the host (MSP) and "draws" them both into memory. These images are then converted to polar format. The feature height data is used to create a shadow mask while the range and azimuth blurring and range attenuation algorithms are applied to the polar SAR data. These are then "anded" together to yield the desired image. The SRSM is controlled by the radar client application (RCA). The RCA determines the mode of operation by setting the SRSM on or off-line and by choosing range setting and display orientation.

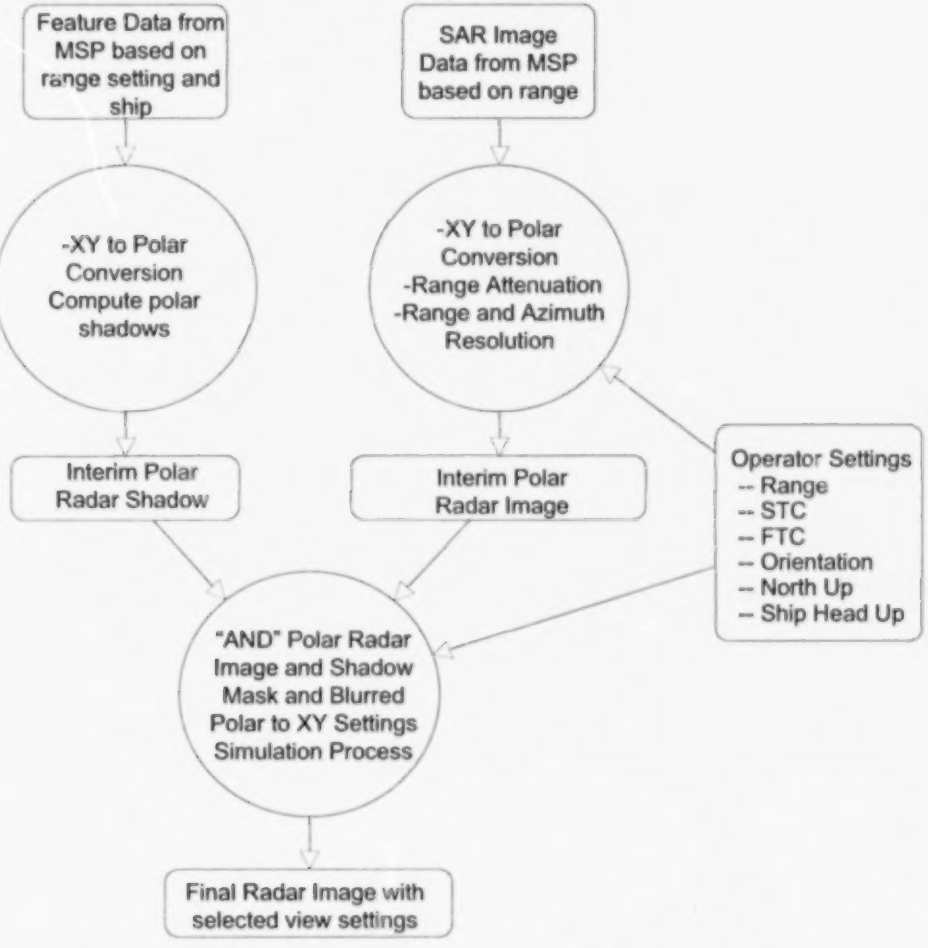

Feature Data from MSP based on range setting and ship
SAR Image Data from MSP based on range
-XY to Polar Conversion Compute polar shadows
-XY to Polar Conversion -Range Attenuation -Range and Azimuth Resolution
Interim Polar Radar Shadow
Interim Polar Radar Image
Operator Settings -- Range -- STC -- FTC -- Orientation -- North Up -- Ship Head Up
"AND" Polar Radar Image and Shadow Mask and Blurred Polar to XY Settings Simulation Process
Final Radar Image with selected view settings

**Figure 6.1-1 SRSM Functional Block Diagram**

## 6.2. SRSM Software Data and Control Flow

Figure 6.2-1 illustrates the basic data and control flow of the SRSM software.

**Figure 6.2-1 The SRSM Data and Control Flow Diagram**

## 6.2. SRSM Software Data and Control Flow

Figure 6.2-1 illustrates the basic data and control flow of the SRSM software.

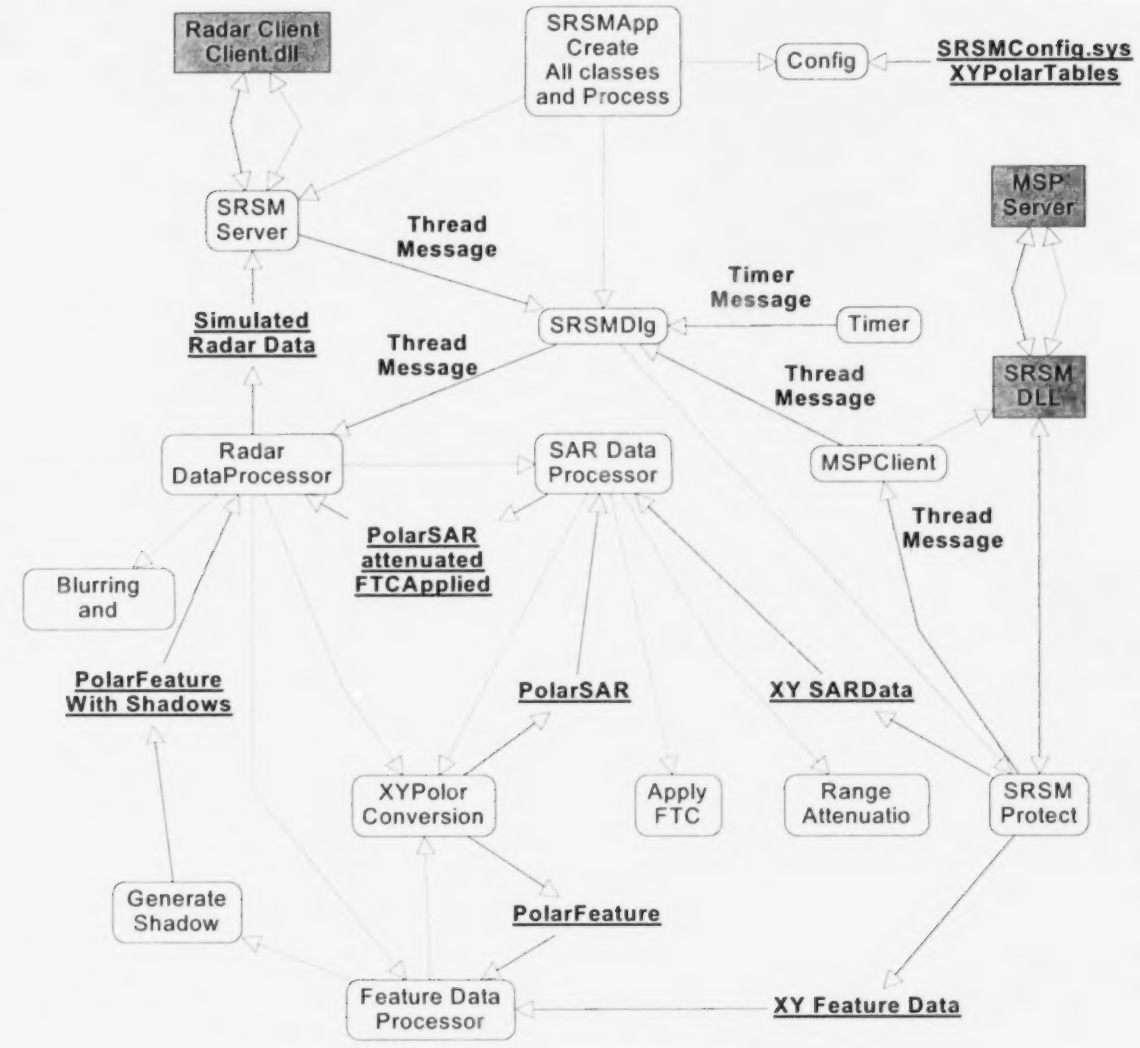



**Figure 6.2-1 The SRSM Data and Control Flow Diagram**

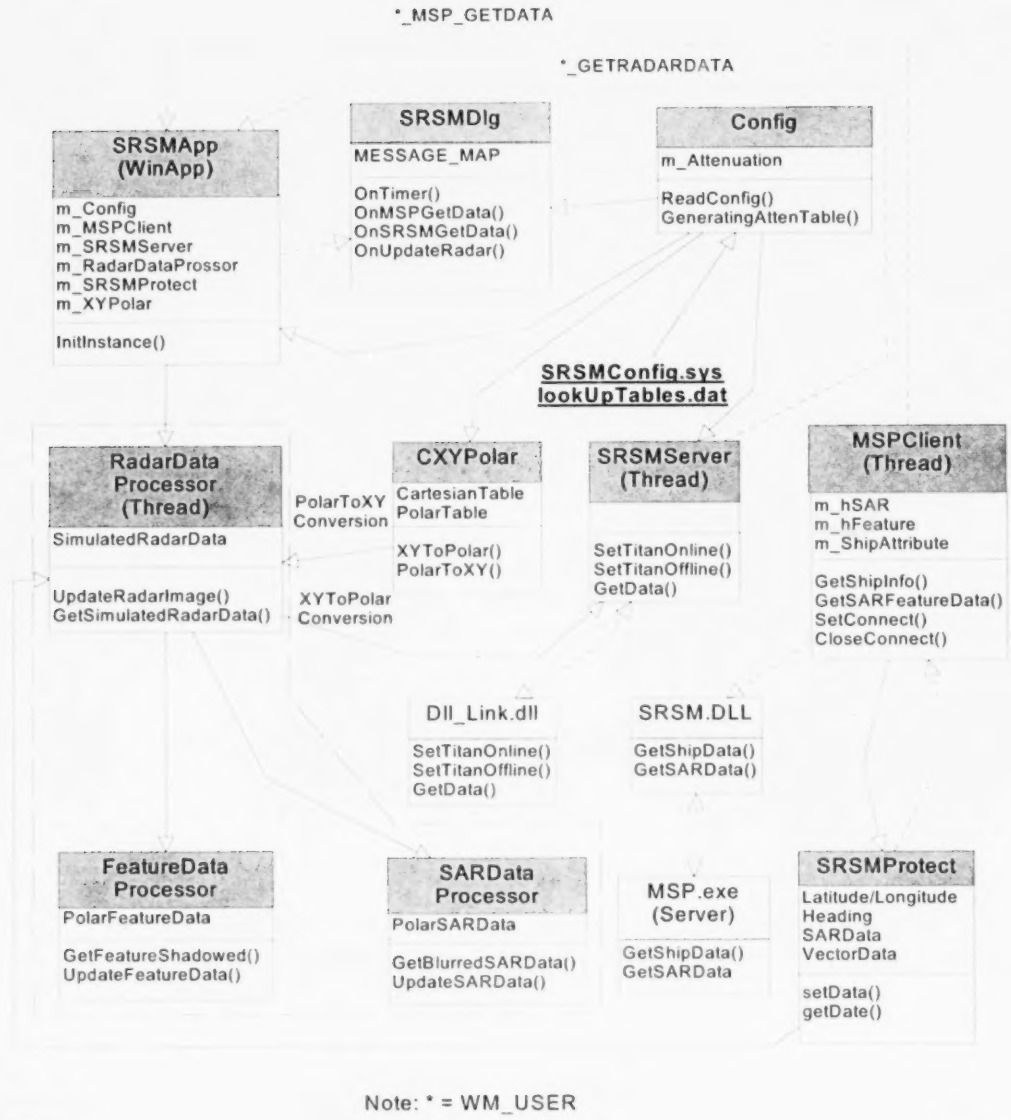
## 6.3. SRSM Software Objects List

The SRSM software is object based. The following is a table of components that make up the SRSM software. Included in Table 6.3-1 is a description of the objects. The tasks or threads are those components that control and react to events in the system. These communication components are DLLs that provide the network interface to the application clients and servers. The data processing classes or objects provide the core of the SRSM application functionality.

**Table 6.3-1 SRM Software Components**

| Object Name | Description |
|---|---|
| **Tasks/Threads** | |
| SRSMApp | This class is used to initialize the SRSM and create the MSPClient, SRSMServer, RadarDataProcessor, SRSMProtect and XY_Polar classes. It contains Config and SRSMDlg classes. |
| SRSMDlg | Controls and synchronizes the tasks in the system by receiving and sending windows notification messages. It also will supply an interface for the user to start, stop or close the application. |
| Config | This class contains all of the initial values that will be read from "SRSMConfig.sys" to start SRSM application. It also builds the attenuation, Radar range index, FTC index tables for RadarDataProcessor class. |
| **Communications** | |
| SRSMServer Thread | SRSM data server for Dll_link.dll. This process contains the server implementation of the remote procedure calls. The parameters are consistent with those of the radar client DLL. |
| DLL_Link.DLL | Radar Client application interface. The routines in this DLL are consistent with the implementation of the MRI. (DLL_link.dll) |
| MSPClient Thread | The class is used to load the SRSM.DLL, set up data communication with MSP by calling the DLL functions. |
| SRSM.DLL | MSP Client Interface. |
| **Application Classes** | |
| SARDataProcessor | Class containing SAR data or "sea clutter" as well as methods to access the SAR data. |
| FeatureDataProcessor | Class containing Feature data and methods to access the feature data. |
| RadarDataProcessor | Class containing both the SARDataProcessor and FeatureDataProcessor classes. It is used to generate a simulated radar image. |
| SRSMProtect | This class is used to control and synchronize access to the memory buffers. All resources such SAR data can only be accessed in sequence. |
| CXYPolar | This class contains the lookup tables and methods used to convert image data between the Cartesian and the polar formats. |

Figure 6.3-1 illustrates the relationship between the various software objects and components that make up the SRSM software.




Note: * = WM_USER

Where:

**Figure 6.3-1 Object Diagram**

## 6.4. SRSM Program File List

Table 6.4-1 outlines the SRSM source code files, with a description of included objects. The SRSM is broken into these different files so that the software can be easily modified and maintained in future.

**Table 6.4-1 SRSM Source Code Files**

| File Name | Objects Included |
|---|---|
| Config.cpp<br>Config.h | Config |
| SRSM.cpp<br>SRSM.h | SRSMApp<br>AboutDlg |
| SRSMDlg.cpp<br>SRSMDlg.h | SRSMDlg |
| SRSMServer.cpp<br>SRSMServer.h | SRSMServer |
| MRISvr_S.c<br>MRISvr.h | Global functions which will be generated by MIDL for RPC communication (Dll_Link) |
| RadarDataProcessor.cpp<br>RadarDataProcessor.h | RadarDataProcessor |
| SARDataProcessor.cpp<br>SARDataProcessor.h | SARDataProcessor |
| FeatureDataProcessor.cpp<br>FeatureDataProcessor.h | FeatureDataProcessor |
| CXYPolar.cpp<br>CXYPolar.h | CXYPolar |
| SRSMProtect.cpp<br>SRSMProtect.h | SRSMProtect |
| MSPClient.cpp<br>MSPClient.h | MSPClient |
| getdata.h | Global functions which will be generated by MIDL for RPC communication (SRSM.DLL) |
| SRSMapi.cpp<br>SRSMapi.h | Global functions which will be used to DIB application such as a paint bitmap on the screen |
| mspcallsdef.h | The definition of the DLL functions call |
| | **Dynamic Link Libraries** |
| SRSMDll.cpp<br>SRSMDl.h | SRSM.DLL |
| Dll_link.dll | Radar Client Functions |

## 6.5. SRSM Operating System

The Shipboard Radar Simulator Module will run under the Microsoft Windows NT 4.0 (Workstation) on a dual Pentium 200 HP Vectra.

## 6.6. SRSM Software Implementation Language

The Implementation Language is C++ and the compiler that will be used is Microsoft Visual C++ 5.0 for Windows NT. This compiler version supports C++ multi-threading and MFC (Microsoft Foundation Classes) 4.2.

# 7. Systems Limits

The system is bounded by the initial SAR data resolution. At lower range settings, pixel replication is very evident. In real radar the resolution is directly related to the beam angle, antenna rotation and is inversely proportional to the target distance. In the SRSM the maximum resolution is determined by the resolution of the source data.

Also the height bitmaps need to be generated in advance of the SRSM server operation. This function just takes too much time to be performed in real time.

# 8. Recommendations and Proposed Improvements

In summary, the SRSM has been successful but requires some additional improvements to increase its performance and to make the radar simulation more realistic. Here is a list of recommendations for improvements. The list is in no particular order.

- Ship's track – The SRSM would be more realistic radar if the ship's track were displayed in the ice field on radar. This would require a dynamic overlay of the ship's track in ice on the simulated image.

- Pixel interpolation synthesis – Interpolating the polar pixels to synthesize additional resolution when range setting is less than 6 nm would improve the image quality. This will remove the pixel replication that occurs on lower range settings.

- X-polarized image synthesis – Currently the SAR data is HH (horizontal transmission/horizontal reception). It could be advantageous to demonstrate how cross-polarized radar can be used as a discriminator between new and multi-year ice.

- Improved sea clutter simulation for open water – At present there is not enough randomness in the clutter. One approach could be the manually generate a series of patterns that are replayed at different intervals.

- Stand-alone image display client – In order to make the SRSM a stand-alone application it might be useful to the user. Also this alleviates the necessity for a third-party software.

- On-line tuning of simulation and display – At present there are no facilities to dynamically tune the radar simulation and display. Items such as beam angle, beam overlap and dynamic data range could all be used enhance the existing model.

- Radar shadow height map data generation – In order to generate realistic radar shadows every pixel of SAR data requires a corresponding height value. If the original SAR data is 200 Mbytes in size, then a corresponding height map of equal value is needed. Currently this takes one to two days depending on size of SAR data and the number of range settings required. Ideally, we would like to optimize this operation.

# 9. Conclusion

Based on the analysis and discussions presented in this report, it is evident that a simulation of shipboard radar is indeed very possible with SAR and Feature files. By applying the principles of radar range attenuation together with range and azimuth blurring to the SAR data and by a careful analysis of height features, it is possible to create realistic marine radar images. In cases where there is no SAR data, it is possible to simulate sea clutter, as experienced in open water navigation. In either situation (with or without SAR) it is possible to process and handle height data such as land outlines, or even isolated ice features, such as icebergs or any other object that has a height property (another ship, for example). The modelling principles outlined in this report can, as a result, form the basis for shipboard radar simulation.

# Appendix A – Technical and User Manual

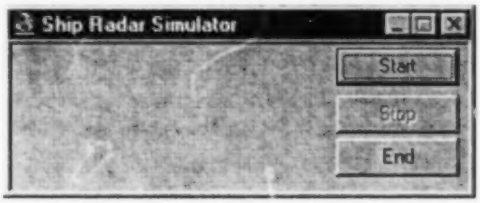
This section describes the Technical and User Manual for the Ice Navigation Simulation – Shipboard Radar Simulation Module Project. The information defined in this document outlines the steps to use the SRSM application.

**Initial Conditions**

It is assumed that the MSP and TSM are operable so that the SRSM can get its data from the MSP. The TSM is required to move the ship. IceNav32 and SetUpMri.exe are used as "Radar Clients" to the SRSM server to show the radar image.

**SRSM User Interface**

The SRSM.exe needs the SRSMConfig.sys to be located in the same directory as the SRSM application in order to be run. Double click the SRSM icon on the window to run the application, the SRSM user interface will pop up on the screen as shown below.




**Start SRSM**

Click the Start button to start getting SAR image, Height Raster and ship information from the MSP. If a message pops up, then the SRSM did not establish a connection with the MSP. The user then has to check whether the MSP is running or whether the value of **$MSPServerName** in the SRSMConfig.sys is correct.

**Stop SRSM**

Click the Stop button to stop getting SAR image, Height Raster and ship information.

**Close SRSM**

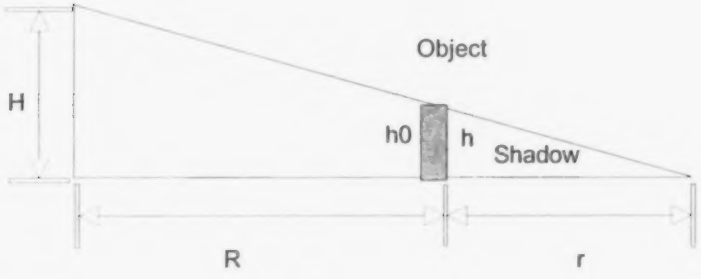
Click the End button to close the SRSM application.

## SRSM Configurability

The SRSMConfig.sys is a text file. The user can use any text editor application to change or modify the configuration values. Changing the values will effect the performance of the SRSM.exe.

**$Radar_Height**

The SRSM will use the Radar_Height value to create the shadow on the Radar Image. If the value is lower, then more shadow will be shown on the image. The following figure shows the relation between the Radar_Height value and the shadow.




Where:

"H" represents the Radar_Height value.

"h0" and "h" represent an object height value.

**$Tables_Directory**

The SRSM needs the **$XYTOPOLAR_TABLE** and **$POLARTOXY_TABLE** tables to translate between the Cartesian and Polar coordinates. This variable contains the directory path of files.

**$Initial_Range**

The **$Initial_Range** value is used to set the initial radar range setting. The units are in nautical miles.

**$MSPServerName**

The **$MSPServerName** variable contains the name of the computer that is running the MSP Server.

**$NBR_RADIALS =1024**

This variable contains the number or radials that make up a single radar sweep. This value should not be changed because the XYToPolar table size is fixed and was created off-line.

**$NBR_PIXELS_PER_RADIAL = 512**

This variable represents the number of pixel samples per radial. This variable should not be changed because the PolarToXY table size is fixed and it was created off-line.

**$IMAGE_HEIGHT&WIDTH = 1024**

This variable represents the final image size. The value should not be changed since the image size on the radar display is fixed to 1024 by 1024.

**$BufferFactor = 150**

In percent, this variable is used to define extended buffer in which to store the SAR image and height raster. This value can't be changed. If the value is changed, then the value at MSP.cfg should be changed too.

**$XYTOPOLAR_TABLE**

This variable represents the XYtoPolar table file name.

**$POLARTOXY_TABLE**

This variable represents the PolartoXY table file name.

**$SRSMDLL_Dir**

This variable indicates the location of the SRSM.DLL. The SRSM application needs the DLL to do the network connection with the MSP so that the SRSM can get data from the MSP.

**$NumberOfModes**

This variable represents the number of the radar range modes (i.e. settings) that will be used in the SRSM.

**$ModeIndexValues =[0.25,0.5,0.75,1.5,3.0,6.0,9.0,12.0, 24.01, 36.02, 48.02]**

This variable represents the list of the Radar ranges and is in Nautical Miles

**Note:**

There are two sets of the $ModeIndexValues in the file. One set is disabled by using "//", the other set is enabled. This will depend on the current "Radar Client" application (IceNav.exe or SetupMri.exe). The set with 20 values is for the SetUpMri.exe. The other is for the IceNav.exe application.

To date, there are two sets of values that cannot be changed because:

- The IceNav32.exe radar display has a fixed range setting. (i.e. hard coded)
- The SetupMri.exe has a fixed range setting. (The index values are fixed with range values)

**$NumberOfFTCs = 8**

This variable is used for the new version of the DLL (dll_link3). The value should not be changed because there are only 8 index values for the FTC setting in SetUpMri.exe.

**$FTCIndexValues =[31.0, 62.0, 93.0, 124.0, 155.0, 186.0, 217.0, 228.0]**

This variable represents the list of the FTC index values.

**$SRSMTimer =6000**

This variable represents the timer in the SRSM. Its unit is milliseconds. The value should be greater than 2000. This is because, for 25 meter SAR data, the ship position cannot change by more than one pixel in 2 second.

# Appendix B – Sample Output Imagery

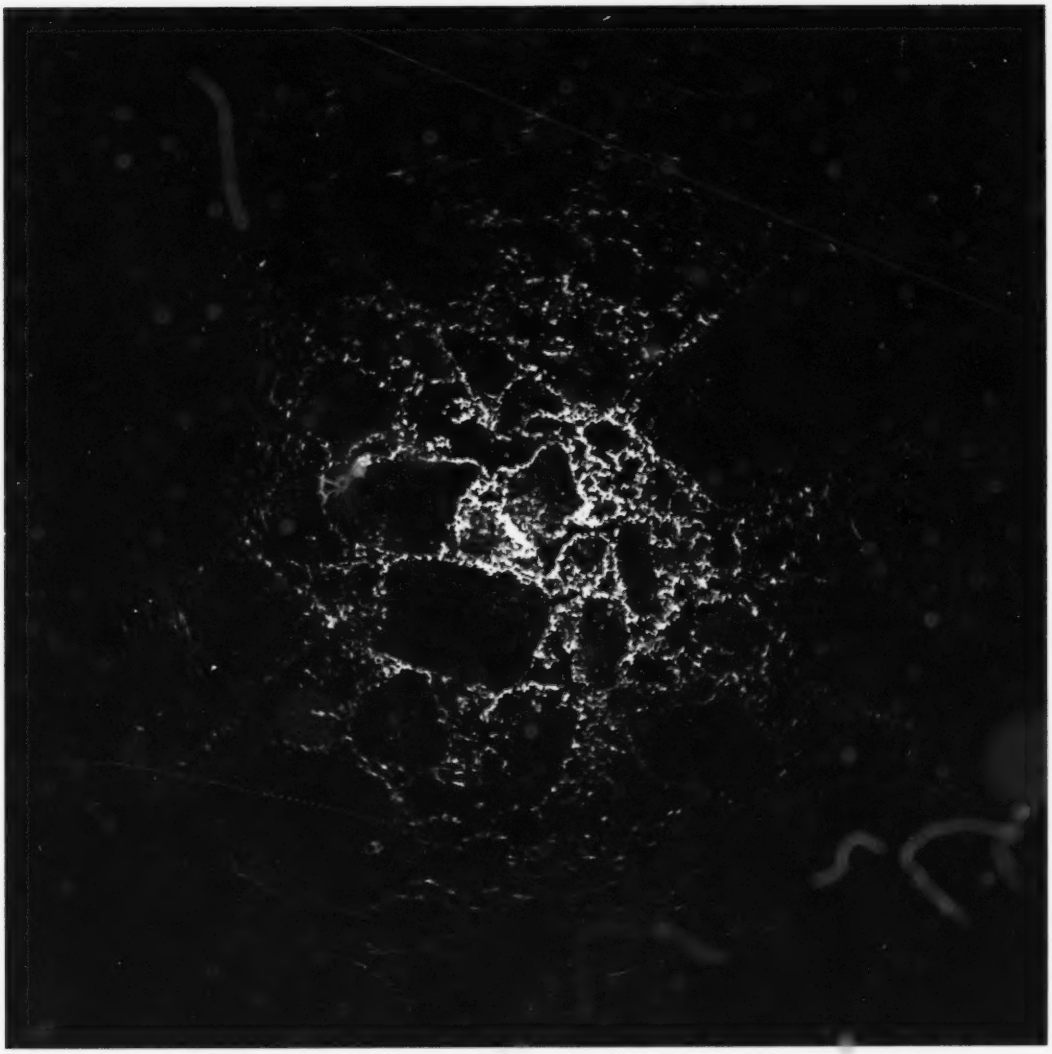


**Figure B-1 Radar Range = 3 NM, OSCFAR Off, Linear Stretch ON (0-75)**

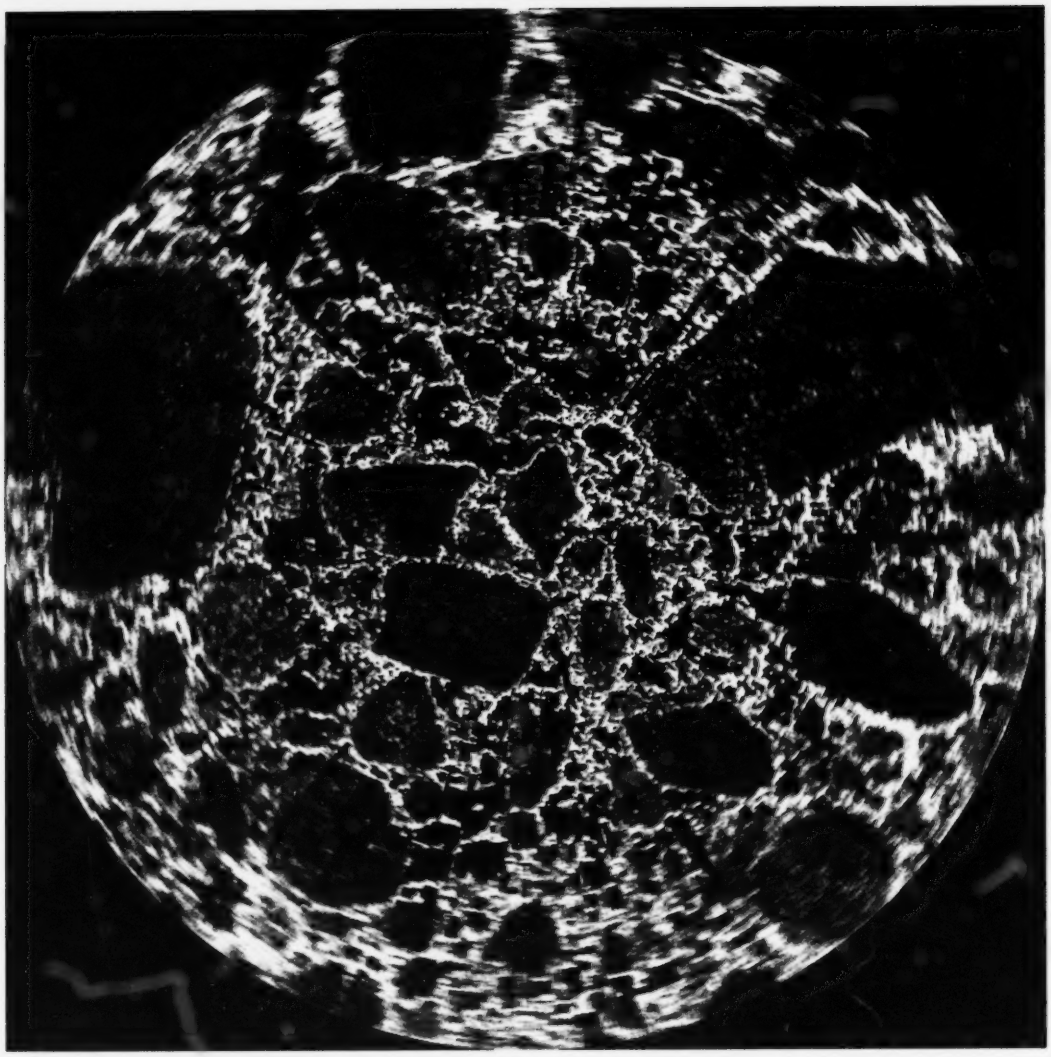


**Figure B-2 Radar Range = 3 NM, OSCFAR ON, Linear Stretch OFF**

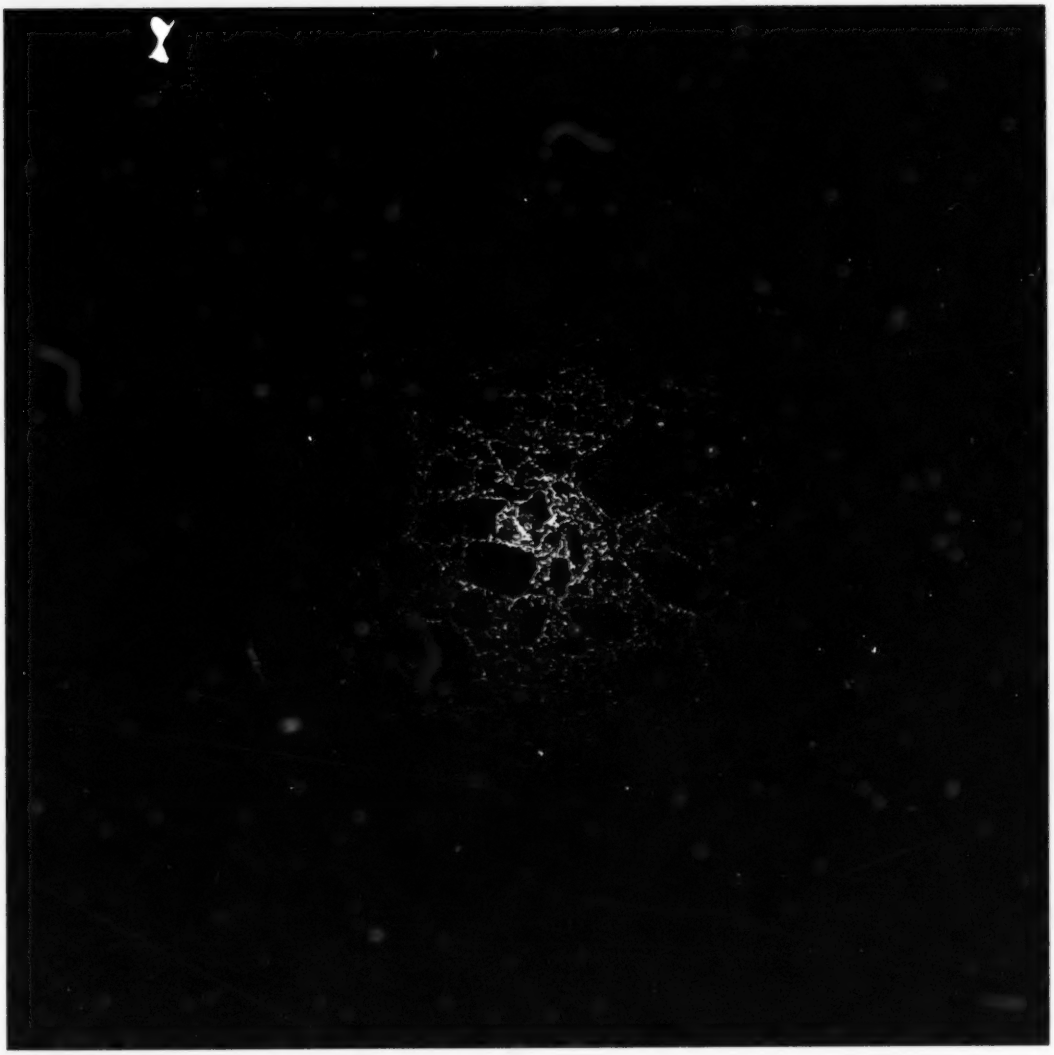


**Figure B-3 Radar Range = 6 NM, OSCFAR Off, Linear Stretch ON (0-75)**

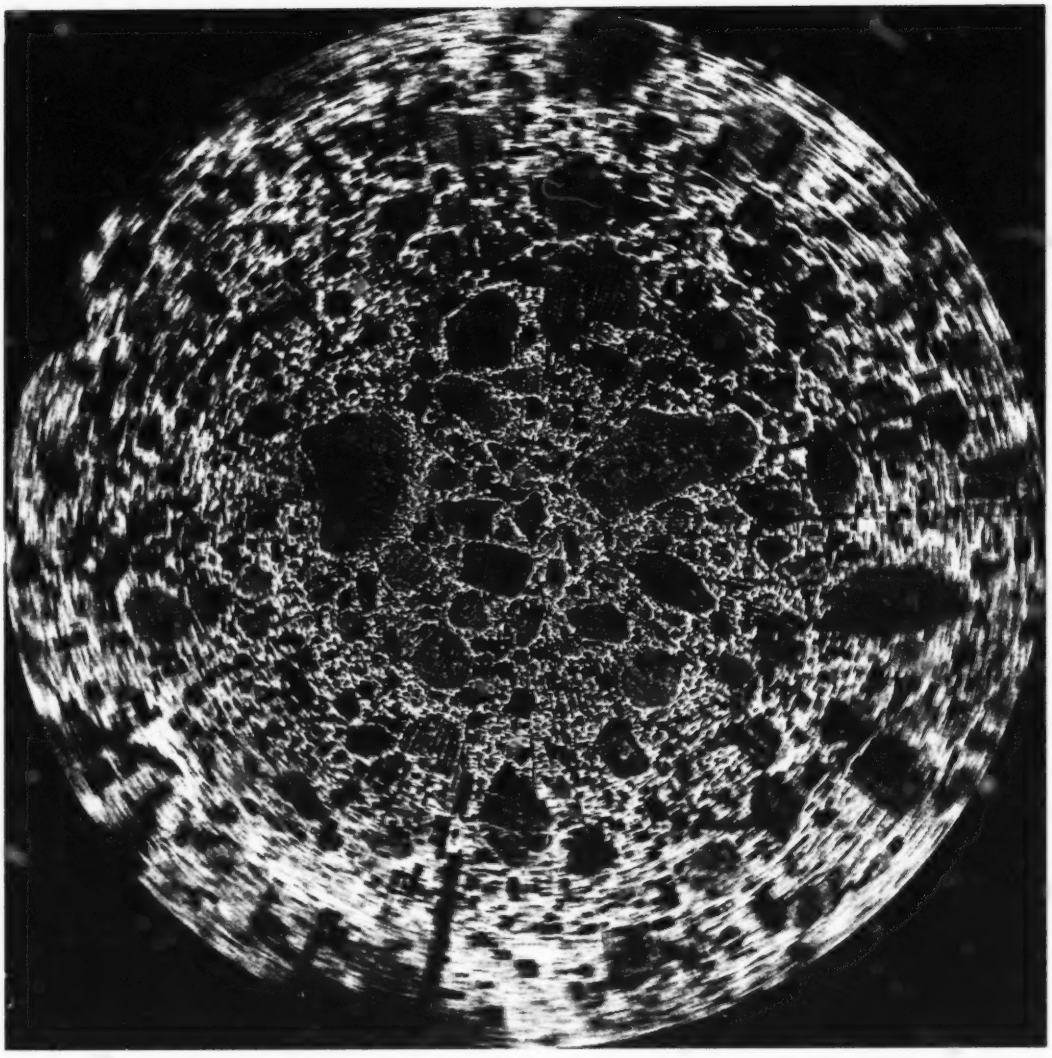


**Figure B-4 Radar Range = 6 NM, OSCFAR ON, Linear Stretch OFF**

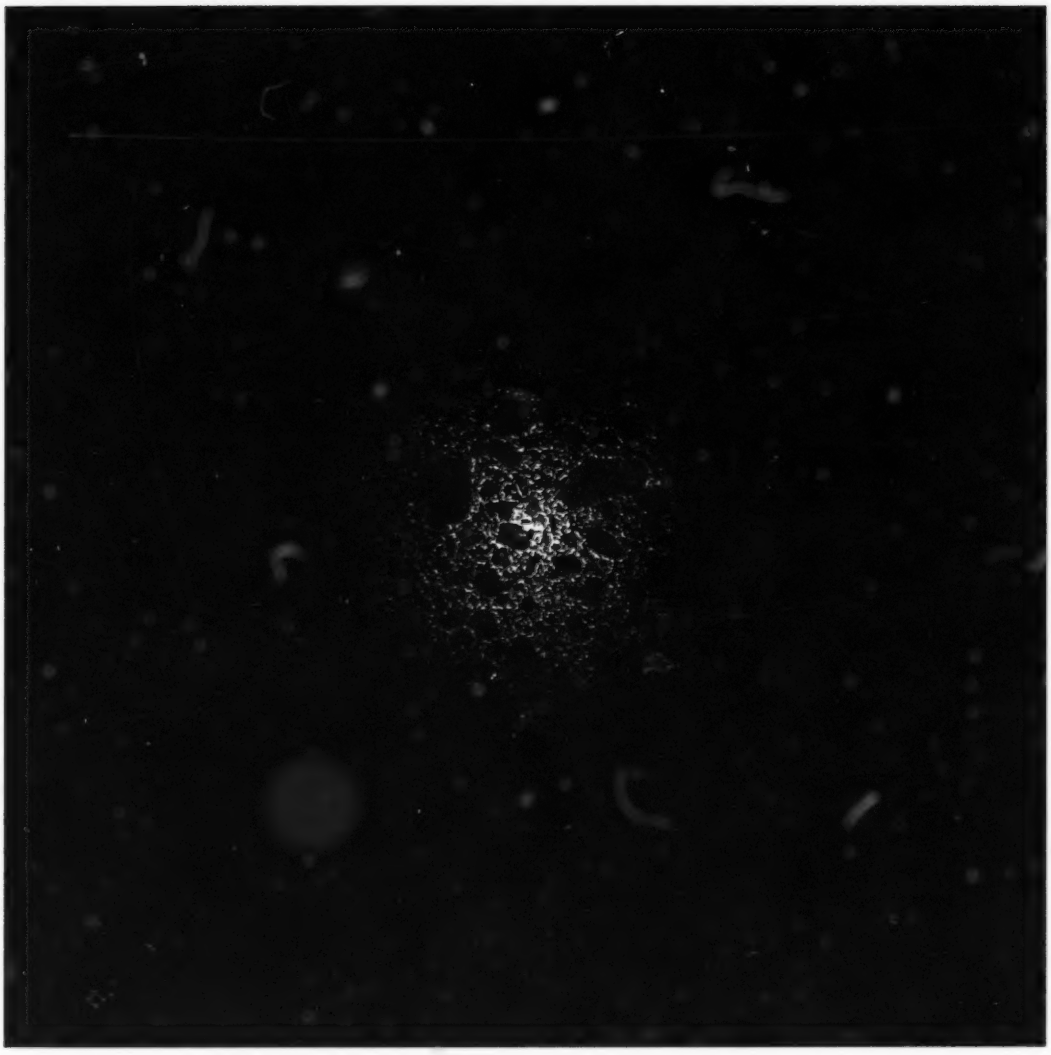


**Figure B-5 Radar Range = 12 NM, OSCFAR Off, Linear Stretch ON (0-75)**

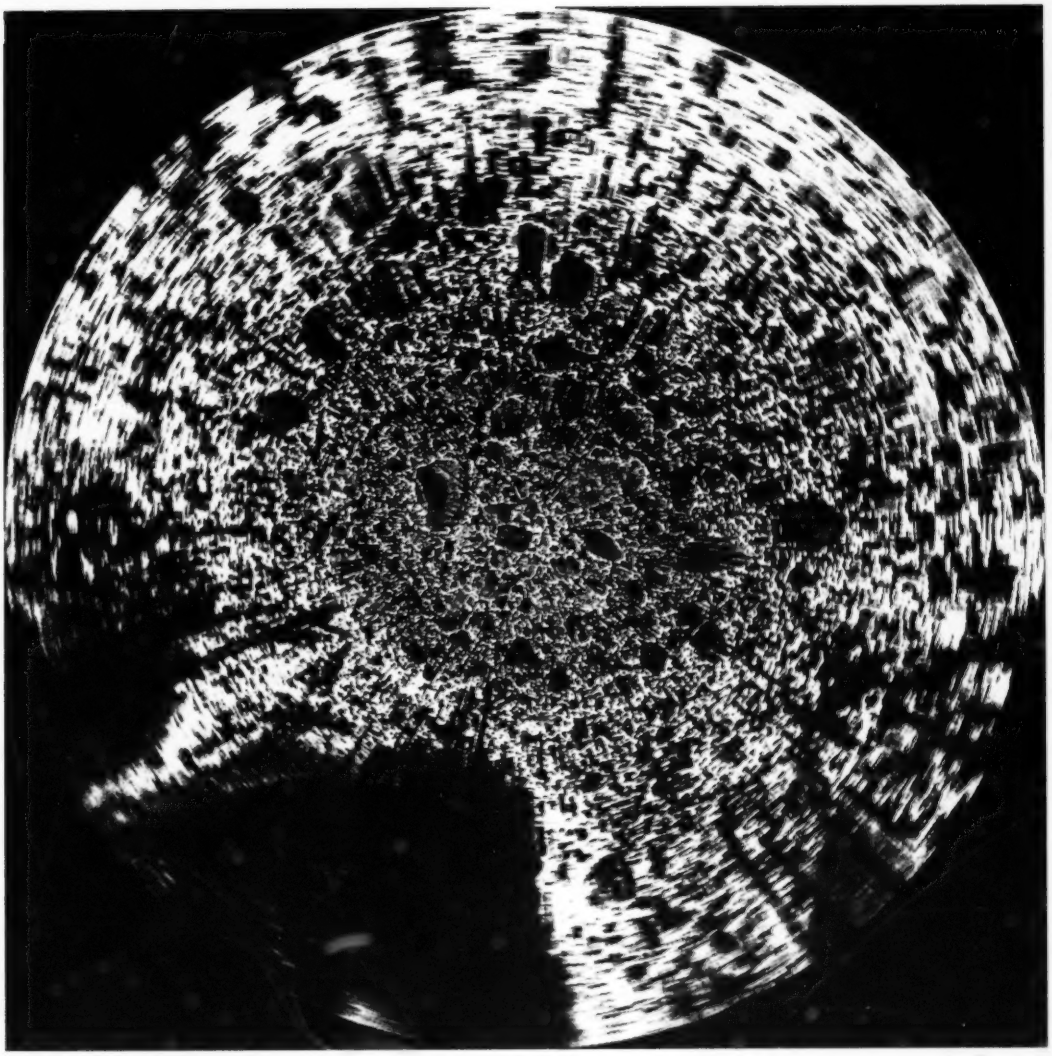


**Figure B-6 Radar Range = 12 NM, OSCFAR ON, Linear Stretch OFF**